REPORT DOCUMENTATION PAGE

AFRL-SR-AR-TR-04-

0313

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
	Final	1 July 2001 - 31 December 2003

4. TITLE AND SUBTITLE
Robust Decision Making: Addressing Uncertainies in Distributions

5a. CONTRACT NUMBER

5b. GRANT NUMBER
F49620-01-1-0338

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Stanislav Uryasev
Panos Pardalos

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Florida
310 Weil Hall
Gainesville, FL 32611-6550

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research
4015 Wilson Blvd
Mail Room 713
Arlington, VA 22203

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Statement A. Approved for public release; Distribution is Unlimited.

13. SUPPLEMENTARY NOTES

20040625 139

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Stanislav Yryasev
U	U	U	UU		19b. TELEPHONE NUMBER *(Include area code)* 352-392-3091

Project Report

Robust Decision Making: Addressing Uncertainties in Distributions

Project #: F49620-01-1-0338

PI: Prof. Stanislav Uryasev
Co-PI: Prof. Panos Pardalos

Risk Management and Financial Engineering Lab
Center for Applied Optimization
Department of Industrial and Systems Engineering
University of Florida, Gainesville, FL, 32611

March 10, 2004

Summary

The project was concentrated on development of new methodologies for decision making in uncertain environment and relevant applications.

The first part of the project was focused on analytical and discrete optimization approaches for routing an aircraft in threat environment. The model considered aircraft trajectory in three-dimensional space. Several threats were studied, including risk of aircraft detection by radars, sensors, and the risk of being killed by surface to air missiles. The problem of finding aircraft optimal risk trajectory subject to a constraint on the trajectory length was solved by analytical and discrete optimization approaches.

The second part of the project resulted in general approach to risk management for the case with uncertainties in distributions. The risk of loss, damage, or failure was measured by the Conditional Value-at-Risk (CVaR) measure. As a function of decision variables, CVaR is convex, and therefore can be efficiently controlled/optimized using convex or linear programming. The methodology was tested on two Weapon-Target Assignment (WTA) problems. The total cost of a mission was minimized, while satisfying the operational constraints and ensuring destruction of targets with high probability. The risk of a failure of the mission is controlled by CVaR constraints. The case studies showed that there were significant qualitative and quantitative differences in solutions of deterministic and stochastic WTA problems.

The third part of the project studied the Multiple Traveling Salesmen Problem (Multiple-TSP) in several variations. The research was focused on MIN-MAX 2-TSP, which cannot be solved by standard methods. The relation between this class of problems and a subclass of the self-dual monotonic Boolean functions was established. This resulted in new efficient optimization algorithms.

Content

4

Part 1: Optimal Path Planning in a Threat Environment

1. Introduction

The class of military and civil engineering applications dealing with optimal trajectory generation for space, air, naval and land vehicles is very broad. It addresses several types of problems with various objectives, constraints on resources and control limitations, for instance,

- *Minimizing risk of aircraft detection by radars, sensors or surface air missiles (SAM) [5, 19, 22]*
- *Minimizing risk of submarine detection by sensors [21]*
- *Minimizing cumulative radiation damage in passing through a contaminated area*
- *Finding optimal trajectories for multiple aircraft avoiding collisions [15]*
- *Maximizing the probability of target detecting by a searcher [1, 3, 9, 12, 13, 16, 17, 20]*
- *Minimizing propellant consumption by a spacecraft in interplanetary and orbit transfers [4]*
- *Minimizing a weighted sum of fuel cost and time cost for a commercial plane*
- *Minimizing energy for a mobile robot on terrains*

We are interested in developing efficient optimization approaches capable of solving a broad class of applications related to trajectory optimization. This chapter, being the first step in accomplishing this task, is primarily focused on optimal path planning for an aircraft in a threat environment. The threat is associated with the risk of aircraft detection by radars, sensors or SAMs. The chapter develops analytical and discrete optimization approaches to optimal trajectory generation that minimize the risk of aircraft detection with: *1) variable aircraft Radar Cross-Section (RCS); 2) different types of detecting installations; 3) arbitrary number of detecting installations; 4) constraint on trajectory length;* and suggests efficient algorithms for solving the formulated risk minimization problem.

Optimal trajectory generation is a fundamental requirement for military aircraft flight management systems. These systems are required to take advantage of all available information in order to perform integrated task processing, reduce pilot workloads and provide updates at regular time intervals sufficient for threat avoidance [19]. A model for optimal routing an aircraft in a threat environment is developed based on specified mission objectives, available resources (fuel capacity), limitations on aircraft control while minimizing risk

exposure. In general, it addresses uncertainty and dynamics inherent to optimal path planning and makes idealizing assumptions with respect to geometrical and physical properties of an aircraft and threat environment. Despite numerous studies in this area, only a few considered risk optimization problems with technological constraints. Zabarankin et al. [22] suggested analytical and discrete optimization approaches for optimal risk path generation in two-dimensional (2D) space with constant RCS, arbitrary number of sensors and a constraint on path length.

This chapter develops a 3D model for minimizing risk of aircraft detection by radars, sensors or SAMs with variable RCS. A sensor is considered to be an antenna capable of receiving an isotropically radiated signal from the aircraft, while a radar is assumed to be an antenna capable of transmitting a signal and receiving the signal reflected off of the aircraft. The model is deterministic and static, since it assumes no uncertainty in aircraft detection and radar locations and considers neither aircraft kinematics equations nor parameters for aircraft control during a flight. The risk of detection is assumed to be independent on aircraft speed. This model extends the 2D risk minimization problem [22] of aircraft detection by sensors to

- *3D space*
- *Variable RCS — an aircraft is considered to be an axisymmetrical ellipsoid with the axis of ellipsoid symmetry determining direction of aircraft trajectory*
- *Risk of detection to be proportional to the aircraft's RCS and reciprocal to the n^{th}-power of the distance between the aircraft and a particular detecting installation, where $n = 2$ corresponds to a passive listener or sensor, and $n = 4$ corresponds to an active listener or radar*

The purpose of this simplified model is analyzing the impact of variable RCS on the 3D geometry of optimal trajectories subject to a constraint on trajectory length and evaluating performance of the developed discrete optimization approach with respect to running time and accuracy. Verified optimization techniques will be applied in optimal path planning with actual-tabulated RCS.

We developed analytical and discrete optimization approaches for solving formulated trajectory optimization problem with a constraint on trajectory length and arbitrary number of detecting installations (sensors or radars). Through techniques of Calculus of Variations, the necessary optimality conditions for a solution to the risk minimization problem were reduced to a nonlinear vectorial differential equation. In the case of a single radar, we obtained an analytical solution to this equation expressed by a quadrature. Analytical solutions are intended for *conceptual understanding and analyzing the impact of variation*

in RCS on the geometry of optimal trajectories and testing performance of the developed discrete optimization approach in the case of a single radar in 2D and 3D spaces. Although we have made significant progress in the development of the analytical approach, finding an analytical solution to the vectorial differential equation in the case of an arbitrary number of installations is still an open issue. This is one of the main reasons for addressing development of discrete optimization approaches.

Several discrete optimization approaches are available for numerical solving proposed risk minimization model. All these approaches may tentatively be divided into three major categories:

- *Gradient-based algorithms*
- *Dynamic programming*
- *Network Flow (NF) optimization*

Efficiency of discrete optimization approaches in optimal risk path planning essentially depends on type of risk functionals, technological constraints, and a scheme of trajectory approximation (see, for instance, [19] for discussions of these issues). Gradient-based algorithms are very efficient when the risk of detection is determined by smooth analytical functionals. However, while dynamic programming and NF optimization are global optimization approaches, gradient-based algorithms most likely find only locally optimal solutions in the case when risk functionals are nonconvex. Many of the previous studies on trajectory generation for military aircraft are concentrated on feasible direction algorithms and dynamic programming [5]. These methods tend to be computationally intense and, therefore, are not well suited for onboard applications. To improve computation time, John and Moore [19] used simple analytical risk functions. Based on such an approach, they developed lateral and vertical algorithms to optimize flight trajectory with respect to time, fuel, aircraft final position, and risk exposure. Nevertheless, these algorithms are not intended for solving optimization problems with technological constraints, such as a constraint on the trajectory length. Zabarankin et al. [22] demonstrated efficiency of NF optimization approach in solving risk minimization problem with a constraint on trajectory length and arbitrary number of sensors in 2D space. The main advantages of using NF optimization approach are

- *Among all feasible approximated trajectories in a considered network, NF approach finds a globally optimal one.*
- *Complexity of NF algorithms is independent on number of detecting installations.*
- *It can easily be applied for the case with actual nonsmooth RCS.*

Despite these advantages, the complexity of NF algorithms substantially depends on the coarseness of a network, in particular, on the number of arcs. Consequently, precision for an optimal solution should reasonably be specified. Recently, Tsitsiklis [18] and Polymenakos et al. [14] suggested Dijkstra-like and correcting-like methods for efficient solving a continuous-space shortest path problem in 2D plane. In this case, finding a globally optimal trajectory employs discretization of Hamilton-Jacobi equation [18], which turns out to be an efficient synthesis of analytical and discrete optimization techniques. This supports the philosophy that using analytical properties of objective functions in NF optimization leads to more efficient algorithms. Since our goal is generating globally optimal trajectories and on the next step applying developed optimization approach in optimal path planning with actual-nonsmooth RCS (in this case utilizing analytical properties of risk functionals is limited), we considered NF optimization approach.

We approximated an admissible domain for aircraft trajectory by a 3D network with a flexible structure and presented aircraft trajectory by a path in this network. NF optimization approach reduced optimal risk path generation with a constraint on trajectory length to the Constrained Shortest Path Problem (CSPP). Development of efficient network structures with relatively small numbers of arcs and nodes while preserving flexibility for trajectory approximation is one of the key issues in reduction of approach computational time.

To solve the CSPP in 2D and 3D cases, we used the Label Setting Algorithm (LSA) with a preprocessing procedure [7, 8] and network structure smoothing. The efficiency of the discrete optimization approach is demonstrated by several numerical examples with various ellipsoid shapes, constraints on trajectory length in the cases of one, two and three radars. For the case with a single radar, we compared analytical and numerical solutions and found that solutions coincide with high precision in 2D case and are very close in 3D case. The fact that discrete trajectories are closer to corresponding analytical ones in 2D case can be explained by different flexibility of 2D and 3D network structures in trajectory approximation. LSA running time in all 2D testing examples is only several seconds, indicating that this NF algorithm is fast enough for use in online applications with a relatively small number of arcs in a graph. However, it is also known that the CSPP is an NP-hard problem and, consequently, no exact polynomial algorithms should be expected. Numerical tests in a 3D case reveal that LSA running time strongly depends on the shape of ellipsoid. This phenomenon has been analyzed from optimization perspective and an improvement for preprocessing procedure has been suggested.

The chapter is organized as follows: section 2 develops a 3D model for trajectory optimization with variable RCS subject to a constraint on trajectory length; section 3 derives the vectorial differential equation for finding the optimal trajectory in a general case and obtains analytical solution to this equa-

tion in the case of a single radar; section 4 reduces optimal path planning to the CSPP and presents the LSA with preprocessing procedure and smoothing condition; section 5 conducts numerical experiments with various ellipsoid shapes and constraints on trajectory length in the cases of one, two and three radars; section 6 analyzes results of numerical experiments from optimization and variable RCS perspectives; section 7 discusses main analytical and numerical results and concludes the chapter; the appendix considers necessary optimal conditions for calculus of variations problem with a nonholonomic constraint and movable end point.

2. Model Development

This section develops a three-dimensional (3D) model for minimizing the risk of aircraft detection by a network of active or passive installations (radars, sensors) with variable aircraft RCS.

Suppose an aircraft must fly from point A, (x_A, y_A, z_A), to point B, (x_B, y_B, z_B), in 3D space trying to minimize the cost of detection from N radars located in the area of interest. We model the aircraft by an axisymmetrical ellipsoid with the axes' lengths a, b and b. The axis with length a is the axis of ellipsoid symmetry, which orients a direction of aircraft trajectory. Ellipsoid shape is defined by parameter $\kappa = b/a$. Cases of $\kappa = 1$, $\kappa < 1$ and $\kappa > 1$ correspond to sphere, elongated and compressed ellipsoids, respectively, see Figure 1.1.

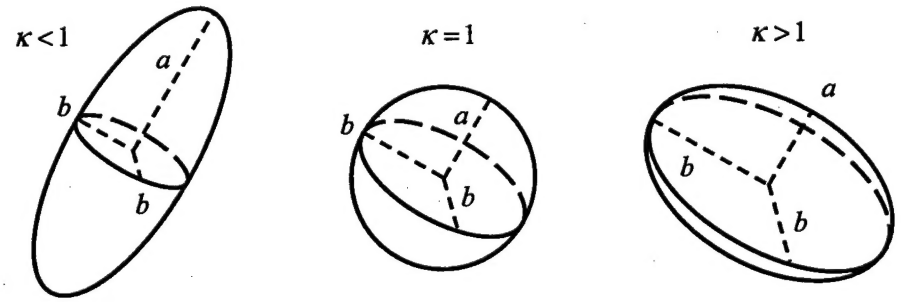

Figure 1.1. Ellipsoid shape is defined by parameter $\kappa = b/a$.

Let vectors $\mathbf{r} = (x, y, z)$ and $\mathbf{q}_i = (a_i, b_i, c_i)$, $i = \overline{1, N}$, determine position of ellipsoid geometrical center and position of the i^{th} radar, respectively. A trajectory of the ellipsoid's center is assumed to be a path of the aircraft. We define a trajectory as a function of its current length s, i.e. $\mathbf{r} =$

$\mathbf{r}(s) = (x(s), y(s), z(s))$. Such a parameterization is also known to be the natural definition of a curve. Vector $\dot{\mathbf{r}}(s) = \frac{d}{ds}\mathbf{r}(s) = (\dot{x}(s), \dot{y}(s), \dot{z}(s))$ determines a direction of aircraft trajectory that coincides with the axis of ellipsoid symmetry. Since $(ds)^2 = (dx)^2 + (dy)^2 + (dz)^2$, vector $\dot{\mathbf{r}}(s)$ must satisfy condition $\dot{\mathbf{r}}^2 = \dot{x}^2 + \dot{y}^2 + \dot{z}^2 = 1$. The length of vector $\mathbf{r}_i(s) = \mathbf{r}(s) - \mathbf{q}_i = (x - a_i, y - b_i, z - c_i)$, denoted by $||\mathbf{r}_i(s)||$, defines the distance from the aircraft to the i^{th} installation (see Figure 1.1), i.e. $||\mathbf{r}_i(s)|| = \sqrt{(x-a_i)^2 + (y-b_i)^2 + (z-c_i)^2}$.

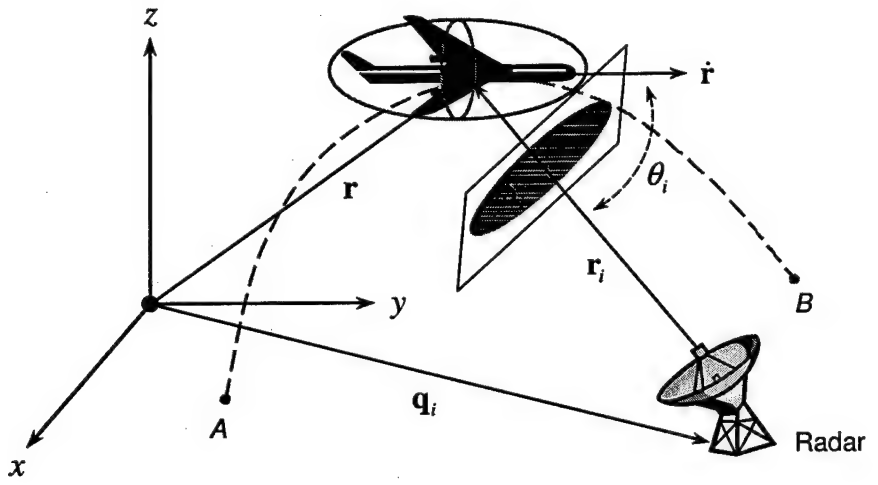

Figure 1.2. 3D model for optimal path planning in a threat environment.

The RCS of the aircraft exposed to the i^{th} radar at point (x, y, z) is proportional to the area of the ellipsoid's projection to the plane orthogonal to vector $\mathbf{r}_i$

$$\text{RCS}_i = \sigma_i S_i,$$

where the constant coefficient σ_i depends on the radar's technical characteristics such as the maximum detection range, the minimum detectable signal, the transmitting power of the antenna, the antenna gain and the wavelength of radar energy.

The magnitude of ellipsoid projection area is given by the formula $S_i = \pi b \sqrt{a^2 \sin^2\theta_i + b^2 \cos^2\theta_i}$, where θ_i is the angle between vectors $\mathbf{r}_i$ and $\dot{\mathbf{r}}$. Based on relation $\cos\theta_i = \frac{\mathbf{r}_i \cdot \dot{\mathbf{r}}}{||\mathbf{r}_i||}$ and using notation $\kappa = b/a$, the formula for RCS_i is identically rewritten as

$$\text{RCS}_i = \sigma_i \pi \left(\frac{a^2+b^2}{2}\right) \frac{2\kappa}{1+\kappa^2} \sqrt{1 + (\kappa^2 - 1)\left(\frac{\mathbf{r}_i \cdot \dot{\mathbf{r}}}{||\mathbf{r}_i||}\right)^2}, \quad k \in [0, +\infty). \tag{1}$$

The purpose of presenting RCS_i in the form of (1) is the following. Since the aircraft has a limited size, we assume the value $\sqrt{a^2+b^2}$ ("diameter" of cross-section) to be constant for all a and b and, hence, the form of the ellipsoid to be defined by ratio b/a, i.e. parameter κ, only. For instance, the case of $\kappa = b/a = 0$ corresponds to an infinitely thin needle with the length of a, while the case of $\kappa = b/a \to \infty$ corresponds to an infinitely thin disk with the radius of b. Note that the cross-section of the infinitely thin needle always equals zero, whereas the cross-section of the infinitely thin disk is reduced to $\sigma_i \pi a^2 |\cos\theta_i|$, which is zero only when $\theta_i = \frac{\pi}{2}$.

The risk function (also referred to as cost function) for detection of the aircraft by the i^{th} radar is proportional to the i^{th} RCS and reciprocal to the n^{th} power of the distance between the aircraft and the i^{th} radar, $||\mathbf{r}_i||^n$ (cases of $n=2$ and $n=4$ correspond to sensor and radar, respectively), namely,

$$C(\mathbf{r}_i, \dot{\mathbf{r}}) = \frac{\mathrm{RCS}_i}{||\mathbf{r}_i||^n} = \sigma_i \pi \left(\frac{a^2+b^2}{2}\right) \frac{2\kappa}{1+\kappa^2} \frac{\sqrt{1+(\kappa^2-1)\left(\frac{\mathbf{r}_i \cdot \dot{\mathbf{r}}}{||\mathbf{r}_i||}\right)^2}}{||\mathbf{r}_i||^n}.$$

Since the value of $\sqrt{a^2+b^2}$ is assumed to be constant, product $\pi\left(\frac{a^2+b^2}{2}\right)$ can be omitted for simplicity and the risk function for detection of the aircraft by the i^{th} installation is reexpressed with normalized coefficient w_i

$$C(\mathbf{r}_i, \dot{\mathbf{r}}) = \frac{2\kappa w_i}{1+\kappa^2} \frac{\sqrt{||\mathbf{r}_i||^2 + (\kappa^2-1)(\mathbf{r}_i \cdot \dot{\mathbf{r}})^2}}{||\mathbf{r}_i||^{n+1}}, \tag{2}$$

where $w_i = \sigma_i \Big/ \sum_{i=1}^{N} \sigma_i$ and $\sum_{i=1}^{N} w_i = 1$.

We assume the risk of detection from N radars at point $\mathbf{r} = (x, y, z)$ to be the sum of risk functions (2) for all $i = \overline{1, N}$

$$L(\mathbf{r}, \dot{\mathbf{r}}) = \sum_{i=1}^{N} C(\mathbf{r}_i, \dot{\mathbf{r}}) = \frac{2\kappa}{1+\kappa^2} \sum_{i=1}^{N} w_i \frac{\sqrt{||\mathbf{r}_i||^2 + (\kappa^2-1)(\mathbf{r}_i \cdot \dot{\mathbf{r}})^2}}{||\mathbf{r}_i||^{n+1}}. \tag{3}$$

The total risk of detection is the integral of (3) along aircraft trajectory with length l, i.e.

$$\mathcal{F}(\mathbf{r}, \dot{\mathbf{r}}) = \int_0^l L\left(\mathbf{r}(s), \dot{\mathbf{r}}(s)\right) ds. \tag{4}$$

The risk minimization problem is finding a trajectory

$$P = \mathbf{r}(s) = (x(s), y(s), z(s)), \; 0 \le s \le l,$$

from point A to point B, having coordinates $\mathbf{r}_A = (x_A, y_A, z_A)$ and $\mathbf{r}_B = (x_B, y_B, z_B)$, respectively, with the minimal risk of detection subject to a constraint on trajectory length

$$\begin{array}{ll} \min\limits_{P} & \mathcal{F}(\mathbf{r}, \dot{\mathbf{r}}) \\ s.\,t. & \dot{\mathbf{r}}^2 = 1, \\ & \mathbf{r}(0) = \mathbf{r}_A,\ \mathbf{r}(l) = \mathbf{r}_B, \\ & l \leq l_*. \end{array} \tag{5}$$

The form of risk functional (4) implies that either the risk is independent on aircraft speed, or aircraft speed is always a unit. Under assumption of unit speed, s becomes time variable t, total length l becomes total time T, and (5) is viewed as a problem of optimal control with $\mathcal{F}(\mathbf{r}, \mathbf{v}) = \int\limits_0^T L\left(\mathbf{r}(t), \mathbf{v}(t)\right)\, dt$, with $\mathbf{v} = \dot{\mathbf{r}}$. Whatever the interpretation of (5) is, analysis techniques are the same. To solve problem (5), calculus of variations and network flow optimization approaches are addressed.

We want to mention the 2D dynamic model for minimizing the risk of submarine detection by a network of sensors [21]. In that model, the risk functional considers different directions of the power radiation and variable speed of a submarine

$$\int\limits_0^T \sum_{i=1}^{N} S_i \frac{(1 - \kappa_2 \cos(2\theta_i(t)))}{||\mathbf{r}_i(t)||^2} \left(1 + \kappa_1 ||\mathbf{r}_t'||^4\right) dt,$$

where $\mathbf{r}_i(t)$ and $\theta_i(t)$ mean exactly the same as in aircraft detection model (see Figure 1.2), i.e. $\mathbf{r}_i(t)$ is the distance between the submarine and the i^{th} sensor and $\theta_i(t)$ is the corresponding angle, which now both depend on time t; $\mathbf{r}_t'$ denotes submarine speed at time moment t; S_i is the sensitivity coefficient associated with the i^{th} sensor; κ_1 and κ_2 are parameters, which correspond to doubling the radiated power relative to small speeds (usually, $\kappa_1 = 0.0003 << 1$) and adjustment of power radiation in different directions ($\kappa_2 \leq 1$; when $\kappa_2 > 0$, power tends to be radiated most strongly broadside; κ_2 is arbitrarily set to 0.5 [21]), respectively. The term $1 - \kappa_2 \cos(2\theta_i(t))$ may be considered as submarine cross-section in 2D space. The Optimal Control approach, suggested to solve the model, starts with some feasible trajectory provided by an observer and transforms it to locally optimal one by steepest-descent technique.

3. Calculus of Variations Approach

This section presents a vectorial differential equation for solving the risk minimization problem (5) and obtains an analytical solution to this problem

in the case of a single radar or sensor. The vectorial differential equation is the reduction of necessary conditions for an extremal minimizing a functional with a nonholonomic constraint and movable end point. Through techniques of Calculus of Variations, the appendix derives this equation in a general case.

Introducing a new function

$$g_i(\mathbf{r}_i, \dot{\mathbf{r}}, \lambda_L) = \frac{1}{||\mathbf{r}_i||^{n-1}\sqrt{||\mathbf{r}_i||^2 + (\kappa^2 - 1)\,(\mathbf{r}_i \cdot \dot{\mathbf{r}})^2}} + \lambda_L, \tag{6}$$

and using notation $\dot{g}_i = \frac{d}{ds} g_i$, we formulate *necessary conditions* for an optimal trajectory satisfying (5).

Theorem 3.1. (vectorial differential equation). *An optimal solution to optimization problem (5) should necessarily satisfy the following vectorial differential equation*

$$\sum_{i=1}^{N} w_i \left(\left(\frac{\mathbf{r}_i}{\mathbf{r}_i \cdot \dot{\mathbf{r}}} - \dot{\mathbf{r}} \right) \dot{g}_i - \ddot{\mathbf{r}}\, g_i \right) = 0, \tag{7}$$

with boundary conditions

$$\mathbf{r}(0) = \mathbf{r}_A, \quad \mathbf{r}(l) = \mathbf{r}_B, \quad l \le l_*, \tag{8}$$

and nonholonomic constraint

$$\phi(\dot{\mathbf{r}}) = \dot{\mathbf{r}}^2 - 1 = 0. \tag{9}$$

Proof. The problem (5) is a particular case of the problem (1.A.1)-(1.A.5) considered in the appendix. In the case of (9) we have $\frac{\partial \phi}{\partial \dot{\mathbf{r}}} = 2\dot{\mathbf{r}}$ and consequently $\frac{\partial \phi}{\partial \dot{\mathbf{r}}} \Big/ \left(\dot{\mathbf{r}} \cdot \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \right) \equiv \dot{\mathbf{r}}$. Substituting the last equality into the general vectorial differential equation (1.A.10) derived in the appendix, we obtain the vectorial differential equation for determining an optimal trajectory $\mathbf{r}(s)$, $0 \le s \le l$,

$$\frac{\partial L}{\partial \mathbf{r}} - \frac{d}{ds}\left(\frac{\partial L}{\partial \dot{\mathbf{r}}} + \dot{\mathbf{r}} \left(L - \dot{\mathbf{r}} \cdot \frac{\partial L}{\partial \dot{\mathbf{r}}} + c_L \right) \right) = 0. \tag{10}$$

Introducing a new constant λ_L by relation $c_L = \frac{2\kappa}{1+\kappa^2}\lambda_L$ and using notations $g_i = g(\mathbf{r}_i, \dot{\mathbf{r}}, \lambda_L)$, $\dot{g}_i = \frac{d}{ds} g\,(\mathbf{r}_i, \dot{\mathbf{r}}, \lambda_L)$, where function $g(\mathbf{r}_i, \dot{\mathbf{r}}, \lambda_L)$ is defined by (6), we verify that the following relations hold for (3)

$$L - \dot{\mathbf{r}} \cdot \frac{\partial L}{\partial \dot{\mathbf{r}}} + c_L = \frac{2\kappa}{1+\kappa^2} \sum_{i=1}^{N} w_i\, g_i,$$

$$\frac{\partial L}{\partial \mathbf{r}} - \frac{d}{ds}\frac{\partial L}{\partial \dot{\mathbf{r}}} = \frac{2\kappa}{1+\kappa^2}\sum_{i=1}^{N} w_i\, \dot{g}_i \left(\frac{\mathbf{r}_i}{\mathbf{r}_i \cdot \dot{\mathbf{r}}}\right),$$

which reduce (10) to equation (7).

Note equation (7) and constraint (9) are dependent in the sense that the scalar product of (7) with $\dot{\mathbf{r}}$ is reduced to $\sum_{i=1}^{N} w_i \left(\left(1-\dot{\mathbf{r}}^2\right)\, \dot{g}_i - \ddot{\mathbf{r}}\, g_i\right) = 0$, which becomes identity if (9) is satisfied. □

Remark: equation (7) may be presented in different forms. Using relation (5), we have $\frac{\mathbf{r}_i}{\mathbf{r}_i\cdot\dot{\mathbf{r}}} - \dot{\mathbf{r}} = \frac{\dot{\mathbf{r}}\times[\mathbf{r}_i\times\dot{\mathbf{r}}]}{\mathbf{r}_i\cdot\dot{\mathbf{r}}}$ and $-\ddot{\mathbf{r}}(\mathbf{r}_i\cdot\dot{\mathbf{r}}) = \dot{\mathbf{r}}\times[\mathbf{r}_i\times\ddot{\mathbf{r}}]$. Consequently, (7) becomes

$$\sum_{i=1}^{N} w_i \left(\frac{\dot{\mathbf{r}}\times[\mathbf{r}_i\times\dot{\mathbf{r}}]}{\mathbf{r}_i\cdot\dot{\mathbf{r}}}\,\dot{g}_i - \ddot{\mathbf{r}}\, g_i\right) = \dot{\mathbf{r}}\times\sum_{i=1}^{N}\frac{w_i}{\mathbf{r}_i\cdot\dot{\mathbf{r}}}\left[\mathbf{r}_i\times\frac{d}{ds}(\dot{\mathbf{r}}\, g_i)\right] = 0.$$

Equation (7) may also be presented in a matrix form. Denoting

$$G_{N\times 1} = \begin{pmatrix} w_1 g_1 \\ w_2 g_2 \\ \vdots \\ w_N g_N \end{pmatrix}, \qquad e_{N\times 1} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix},$$

$$R_{3\times N} = \left(\frac{\mathbf{r}_1}{\mathbf{r}_1\cdot\dot{\mathbf{r}}}, \frac{\mathbf{r}_2}{\mathbf{r}_2\cdot\dot{\mathbf{r}}}, \cdots, \frac{\mathbf{r}_N}{\mathbf{r}_N\cdot\dot{\mathbf{r}}}\right),$$

(7) is rewritten as

$$R\dot{G} = \frac{d}{ds}\left(\dot{\mathbf{r}}(e^{\top}\cdot G)\right).$$

Choice of a form for (7) is just a matter of convenience for conducting analytical manipulations or numerical analysis. Differential equation (7) may be solved numerically by an appropriate gradient-based algorithm, however, in this case we are not guaranteed to obtain a globally optimal solution.

Deriving an analytical solution to (7)–(9) with an arbitrary number of radars or sensors is reduced to finding the second integral for equation (7) (the first one is nonholonomic constraint (9)), which still remains an open issue. The next theorem shows how the second integral and a corresponding analytical solution are found in the case of a single radar or sensor.

Theorem 3.2. (the case of a single radar or sensor). *In the case of a single radar or sensor, located at the origin of the system of coordinates, i.e. point* $(0,0,0)$,

(1) *the optimal trajectory is a planar curve in 3D space, where the trajectory's plane is determined by the origin of the system of coordinates and the starting and finishing trajectory's points, i.e. by* $(0,0,0)$, $\mathbf{r}_A$ *and* $\mathbf{r}_B$ *(the equation of the plane is given by* $[\mathbf{r}_A \times \mathbf{r}_B] \cdot \mathbf{r} = 0$*);*

(2) *introducing a polar system coordinates* (ρ, ψ) *in the trajectory's plane, vectorial differential equation (7) with (8) and (9) is reduced to a nonlinear first-order differential equation with respect to function* $\rho = \rho(\psi)$

$$\frac{1}{\rho^{n-2}\sqrt{\kappa^2(\rho'_\psi)^2 + \rho^2}} + \frac{\lambda_L \rho^2}{\sqrt{(\rho'_\psi)^2 + \rho^2}} = C, \tag{11}$$

with boundary conditions

$$\rho(\psi_A) = \rho_A, \quad \rho(\psi_B) = \rho_B, \tag{12}$$

defining points A and B in the polar system (ρ, ψ)*, and a constraint on trajectory length*

$$\int_{\psi_A}^{\psi_B} \sqrt{(\rho'_\psi)^2 + \rho^2}\, d\psi = l_*. \tag{13}$$

Proof. Since an analytical solution to (7) is derived in the case of a single radar, without loss of generality, we assume that the radar is located at the origin of the system of coordinates, that is, $(a_1, b_1, c_1) = (0,0,0)$, and $\mathbf{r}_1 = \mathbf{r}$. Functions $L(\mathbf{r}, \dot{\mathbf{r}})$, $g(\mathbf{r}, \dot{\mathbf{r}}, \lambda_L)$ and equation (7) in this case are presented, respectively,

$$L(\mathbf{r}, \dot{\mathbf{r}}) = \frac{2\kappa}{1+\kappa^2} \frac{\sqrt{||\mathbf{r}||^2 + (\kappa^2 - 1)(\mathbf{r} \cdot \dot{\mathbf{r}})^2}}{||\mathbf{r}||^{n+1}},$$

$$g(\mathbf{r}, \dot{\mathbf{r}}, \lambda_L) = \frac{1}{||\mathbf{r}||^{n-1}\sqrt{||\mathbf{r}||^2 + (\kappa^2 - 1)(\mathbf{r} \cdot \dot{\mathbf{r}})^2}} + \lambda_L,$$

$$\left(\frac{\mathbf{r}}{\mathbf{r} \cdot \dot{\mathbf{r}}} - \dot{\mathbf{r}}\right) \dot{g} - \ddot{\mathbf{r}}\, g = 0.$$

Producing *vectorial* product of the last equation with vector $\mathbf{r}$, we obtain

$$\frac{d}{ds}\left([\mathbf{r} \times \dot{\mathbf{r}}]\, g\right) = 0,$$

which is equivalent to having the first integral

$$[\mathbf{r} \times \dot{\mathbf{r}}]\, g = \mathbf{C}, \tag{14}$$

where $\mathbf{C} = (C_1, C_2, C_3)$ is a constant vector. Since $(\mathbf{r} \cdot [\mathbf{r} \times \dot{\mathbf{r}}]) = 0$ and $g(\mathbf{r}, \dot{\mathbf{r}}, \lambda_L) \not\equiv 0$, the *scalar* product of (14) with $\mathbf{r}$ leads to

$$\mathbf{C} \cdot \mathbf{r} = 0,$$

which is the equation of a plane going through the origin of the system of coordinates. It means that an optimal trajectory is a *planar* curve in a 3D space, i.e. all its points form a single plane in a 3D space (lie within the same plane). Since boundary points A and B must also belong to trajectory's plane, i.e. vectors $\mathbf{r}_A$, $\mathbf{r}_B$ must satisfy equation $\mathbf{C} \cdot \mathbf{r} = 0$, vector $\mathbf{C}$ is parallel to $[\mathbf{r}_A \times \mathbf{r}_B]$, and the explicit expression for the trajectory's plane is given by

$$[\mathbf{r}_A \times \mathbf{r}_B] \cdot \mathbf{r} = 0, \tag{15}$$

or

$$h_x\, x + h_y\, y + h_z\, z = 0,$$

where (h_x, h_y, h_z) are the components of vector $[\mathbf{r}_A \times \mathbf{r}_B]$

$$(h_x, h_y, h_z) = (y_A z_B - y_B z_A,\ z_A x_B - z_B x_A,\ x_A y_B - x_B y_A).$$

The next step is parameterizing 3D plane (15) by a 2D polar system of coordinates (ρ, ψ). A point with coordinates $(x(\rho,\psi), y(\rho,\psi), z(\rho,\psi))$ should satisfy (15) identically. Suppose the origin of the polar system (ρ, ψ) coincides with the origin of the original 3D system of coordinates, i.e. point $(0,0,0)$. Let ψ be a counterclockwise angle producing left-handed screw with the vector $[\mathbf{r}_A \times \mathbf{r}_B]$ and counted from the upper side of the plane xy. Introducing notations

$$\cos\alpha = \frac{h_x}{\sqrt{h_x^2+h_y^2}}, \quad \cos\beta = \frac{h_z}{\sqrt{h_x^2+h_y^2+h_z^2}},$$
$$\sin\alpha = \frac{h_y}{\sqrt{h_x^2+h_y^2}}, \quad \sin\beta = \frac{\sqrt{h_x^2+h_y^2}}{\sqrt{h_x^2+h_y^2+h_z^2}},$$

coordinates (x, y, z) of points identically satisfying (15) are determined by the following relations

$$\begin{aligned} x(\rho,\psi) &= \rho\ (\sin\alpha\ \cos\psi - \cos\alpha\ \cos\beta\ \sin\psi), \\ y(\rho,\psi) &= -\rho\ (\cos\alpha\ \cos\psi + \sin\alpha\ \cos\beta\ \sin\psi), \\ z(\rho,\psi) &= \rho\ \sin\beta\ \sin\psi. \end{aligned}$$

Based on these relations, we have

$$[\mathbf{r} \times \dot{\mathbf{r}}] = -\rho^2 \dot{\psi} \frac{[\mathbf{r}_A \times \mathbf{r}_B]}{\|[\mathbf{r}_A \times \mathbf{r}_B]\|},$$

and, consequently, using the last formula, (14) is reduced to the following scalar equation

$$\rho^2 \dot{\psi}\, g = C, \tag{16}$$

where C is unknown constant scalar value. Since $\mathbf{r}^2 = ||\mathbf{r}||^2 = \rho^2$, function $g(\mathbf{r}, \dot{\mathbf{r}}, \lambda_L)$ is rewritten as

$$g(\rho, \dot{\rho}, \lambda_L) = \frac{1}{\rho^n \sqrt{1 + (\kappa^2 - 1)\dot{\rho}^2}} + \lambda_L.$$

With this relation, equation (16) and constraint (9), expressed in terms of (ρ, ψ) as $\dot{\rho}^2 + \rho^2\dot{\psi}^2 = 1$, determine a system of differential equations for finding optimal $\rho(s)$ and $\psi(s)$

$$\left(\frac{1}{\rho^{n-2}\sqrt{1 + (\kappa^2 - 1)\dot{\rho}^2}} + \lambda_L \rho^2\right)\dot{\psi} = C, \qquad \dot{\rho}^2 + \rho^2\dot{\psi}^2 = 1, \quad (17)$$

with boundary conditions $\rho(0) = \rho_A$, $\psi(0) = \psi_A$, $\rho(l_*) = \rho_B$, $\psi(l_*) = \psi_B$, where (ρ_A, ψ_A) and (ρ_B, ψ_B) are given by

$$\rho_A = ||\mathbf{r}_A||, \quad \psi_A = \arccos\left(\frac{x_A \sin\alpha - y_A \cos\alpha}{||\mathbf{r}_A||}\right),$$
$$\rho_B = ||\mathbf{r}_B||, \quad \psi_B = \arccos\left(\frac{x_B \sin\alpha - y_B \cos\alpha}{||\mathbf{r}_B||}\right).$$

Let $\rho'_\psi = \frac{d\rho}{d\psi}$. Using relation $\dot{\rho} = \rho'_\psi \dot{\psi}$ with the second equation of (17) we present $\dot{\rho}$ and $\dot{\psi}$ as

$$\dot{\rho} = \pm\frac{\rho'_\psi}{\sqrt{(\rho'_\psi)^2 + \rho^2}}, \qquad \dot{\psi} = \pm\frac{1}{\sqrt{(\rho'_\psi)^2 + \rho^2}}.$$

Substitution of the last formulas into the first equation of (17) eliminates variable s from the system (17) and reduces it to the nonlinear first-order differential equation (11) determining ρ as a function of ψ with boundary conditions (12). Since variable s was eliminated from (17), the second equation of (17) is satisfied identically, and, thus, a constraint on trajectory length should be included in the form of (13).

Note it does not matter what sign, plus or minus, we choose for $\dot{\psi}$ in $\dot{\psi} = \pm\frac{1}{(\rho'_\psi)^2 + \rho^2}$, since we always can change the sign of the constant C in the right-hand side of equation (14) and denote it by a new constant. □

Discussion of necessary and sufficient conditions for a minimum. Equation (11) (or system (17)) is only the necessary condition for a trajectory to be just an extremal, since (11) finds trajectories minimizing the functional (4) under given conditions as well as maximizing it. *A sufficient condition* for a solution to system (11) to be an optimal trajectory (ρ_*, ψ_*), i.e.

to minimize the functional, requires the second variation of the functional at (ρ_*, ψ_*) to be greater than or equal to zero (see the appendix).

In the case of a single radar, the risk functional with the relaxed constraint $\dot{\rho}^2 + \rho^2\dot{\psi}^2 = 1$ is presented in polar coordinates by

$$\mathcal{F} = \int_0^l L(\rho, \dot{\rho}, \dot{\psi})ds,$$

$$L(\rho, \dot{\rho}, \dot{\psi}) = \frac{2\kappa}{1+\kappa^2}\frac{\sqrt{1+(\kappa^2-1)\dot{\rho}^2}}{\rho^n} + \lambda^*\left(\dot{\rho}^2 + \rho^2\dot{\psi}^2 - 1\right).$$

where $\lambda^* = \frac{\kappa}{1+\kappa^2}\left(\frac{1}{\rho_*^n\sqrt{1+(\kappa^2-1)\dot{\rho}_*^2}} + \lambda_L\right) = \frac{\kappa}{1+\kappa^2}\,g^*$. Assuming the constraint on the length of a trajectory to be active, i.e. $l = l_*$, the second variation of the functional $\mathcal{F}$ at (ρ_*, ψ_*) is defined

$$\begin{aligned}
\delta^2\mathcal{F} &= \int_0^{l_*}\Big(L''_{\rho\rho}(\delta\rho)^2 + L''_{\dot{\rho}\dot{\rho}}(\delta\dot{\rho})^2 + L''_{\dot{\psi}\dot{\psi}}(\delta\dot{\psi})^2 + 2L''_{\rho\dot{\rho}}\delta\rho\ \dot{\rho} \\
&\quad + 2L''_{\rho\dot{\psi}}\delta\rho\ \dot{\psi} + 2L''_{\dot{\rho}\dot{\psi}}\delta\dot{\rho}\delta\dot{\psi}\Big)\Big|_{\rho=\rho^*,\,\psi=\psi^*} ds \\
&= \int_0^{l_*}\left(\left(L''_{\rho\rho} - \tfrac{d}{ds}L''_{\rho\dot{\rho}}\right)(\delta\rho)^2 + L''_{\dot{\rho}\dot{\rho}}(\delta\dot{\rho})^2 + L''_{\dot{\psi}\dot{\psi}}(\delta\dot{\psi})^2\right. \\
&\quad \left. + 2L''_{\rho\dot{\psi}}\delta\rho\ \dot{\psi} + 2L''_{\dot{\rho}\dot{\psi}}\delta\dot{\rho}\delta\dot{\psi}\right)\Big|_{\rho=\rho^*,\,\psi=\psi^*} ds \\
&= \tfrac{2\kappa}{1+\kappa^2}\int_0^{l_*}\left(P(\delta\rho)^2 + Q(\delta\dot{\rho})^2 + g^*\rho_*^2(\delta\dot{\psi})^2 + 4g^*\rho_*\dot{\psi}_*\delta\rho\ \dot{\psi}\right) ds \\
&= \tfrac{2\kappa}{1+\kappa^2}\int_0^{l_*}\left(\left(P - 4g^*\dot{\psi}_*^2\right)(\delta\rho)^2 + Q(\delta\dot{\rho})^2\right. \\
&\quad \left. + g^*\left(2\dot{\psi}_*\delta\rho + \rho_*\delta\dot{\psi}\right)^2\right) ds,
\end{aligned}$$

where P and Q are given by

$$P = -\frac{n}{\dot{\rho}_*}\frac{d}{ds}\left(\frac{1}{\rho_*^{n+1}\sqrt{1+(\kappa^2-1)\dot{\rho}_*^2}}\right),$$

$$Q = \frac{\kappa^2-1}{\rho_*^n\left(1+(\kappa^2-1)\dot{\rho}_*^2\right)^{\frac{3}{2}}} + g^*.$$

Since the extremal (ρ_*, ψ_*) satisfies (17), we can use (17) to rearrange P, Q and the other terms in the integral of $\delta^2\mathcal{F}$ and, thus, obtain different equivalent expressions for $\delta^2\mathcal{F}$. However, verification of the condition $\delta^2\mathcal{F} \geq 0$ for all $\delta\rho$, $\delta\dot{\rho}$ and $\delta\dot{\psi}$, even in this particular case, is not a trivial task.

We confine ourselves here only to verification of necessary conditions for a minimum. For an extremal (ρ_*, ψ_*) to minimize the functional, it should necessarily satisfy *the Legendre conditions*

$$L''_{\dot\rho\dot\rho}\Big|_{\rho=\rho^*,\,\psi=\psi^*} \geq 0, \qquad \begin{vmatrix} L''_{\dot\rho\dot\rho} & L''_{\dot\rho\dot\psi} \\ L''_{\dot\rho\dot\psi} & L''_{\dot\psi\dot\psi} \end{vmatrix}_{\rho=\rho^*,\,\psi=\psi^*} \geq 0,$$

which are reduced to verification of $Q \geq 0$ and $g^* \geq 0$. In the case of $k \geq 1$, condition $\lambda_L \geq 0$ guarantees satisfaction of both $Q \geq 0$ and $g^* \geq 0$, which, however, may not be sufficient to guarantee $Q \geq 0$ when $k < 1$. The assumption of λ_L's positivity will play a crucial role in finding appropriate values of λ_L and C in numerical examples.

Although an analytical verification of whether a particular extremal trajectory minimizes the functional is cumbersome, the graph of this trajectory immediately reveals what kind of an extremal it is. Indeed, if the line passing through points A and B separates the trajectory and the radar/sensor, i.e. the trajectory moves away from a detecting installation (the trajectory is *"concave"*), then it minimizes the risk, and vise-versa, if the trajectory moves towards the detecting installation (the trajectory is *"convex"*), then it maximizes the risk.

An analytical solution to equation (11) with boundary conditions (12) and constraint (13) is presented in the next theorem.

Theorem 3.3. (analytical solution in the case of a single radar or sensor). *An analytical solution for nonlinear first-order differential equation (11) with conditions (12), (13) is given by the following quadrature*

$$\psi(\rho) = \psi_A \pm \int_{\rho_A}^{\rho} \frac{d\tau}{\sqrt{(\upsilon^*(\tau, \lambda_L, C))^2 - \tau^2}}, \tag{18}$$

where $\upsilon^*(\rho, \lambda_L, C)$ is a positive root of the following algebraic equation (quartic equation)

$$f(\upsilon) = \rho^{n-2}\left(C\upsilon - \lambda_L\rho^2\right)\sqrt{\kappa^2\upsilon^2 + (1-\kappa^2)\rho^2} - \upsilon = 0, \tag{19}$$

and unknown constants λ_L and C are found from the conditions

(1) $\int_{\psi_A}^{\psi_B} \upsilon^*(\rho, \lambda_L, C)\, d\psi = l_*$ *and* $\psi(\rho_B) = \psi_B$ *if the length constraint is active;*

(2) $\lambda_L = 0$ *and* $\psi(\rho_B) = \psi_B$ *if the length constraint is inactive.*

Proof. The main technique for solving any first-order differential equation analytically is to explicitly express the derivative of an unknown function. By

introducing an auxiliary function

$$\upsilon = \sqrt{(\rho'_\psi)^2 + \rho^2}, \tag{20}$$

we reduce (11) to the algebraic equation (19) with respect to υ, which is a particular case of the following quartic equation

$$\rho^{2(n-2)} \left(C\upsilon - \lambda_L \rho^2\right)^2 \left(\kappa^2 \upsilon^2 + (1-\kappa^2)\rho^2\right) - \upsilon^2 = 0.$$

Explicit analytical expressions for four roots of any quartic equation may be presented by Cardan's (Cardan-Ferrari's) formulas. This is a crucial point in obtaining an analytical solution for the differential equation (11). Due to the cumbersome form of the expressions for the roots of equation (19) we do not present them here. Suppose that $\upsilon^*(\rho, \lambda_L, C)$ is a root for (19), then according to (20), derivative ρ'_ψ is expressed

$$\rho'_\psi = \pm\sqrt{\left(\upsilon^*(\rho, \lambda_L, C)\right)^2 - \rho^2},$$

which leads to a quadrature expression for $\psi = \psi(\rho)$

$$\psi(\rho) = \pm \int \frac{d\rho}{\sqrt{\left(\upsilon^*(\rho, \lambda_L, C)\right)^2 - \rho^2}} + D.$$

Excluding constant D based on boundary conditions $\psi(\rho_A) = \psi_A$ and $\psi(\rho_B) = \psi_B$, this quadrature is reduced to the form of (18).

Note a root for (19) depends on values of λ_L and C. Which root should be chosen with respect to λ_L and C and what are the estimates for λ_L and C are the subject of the next theorem. □

The quadrature (18) is considered to be an analytical solution, since the roots of the quartic equation (19) may be expressed by Cardan's (Cardan-Ferrari's) formulas analytically. There are two special cases when the quadrature (18) is simplified.

Example 1 (the optimization problem without a constraint on trajectory length). The first case corresponds to the optimization problem without a constraint on trajectory length, in this case an optimal trajectory is presented by Rhodenea (rose function)

$$\rho(\psi) = C^{-\frac{1}{n-1}} \sin^{\frac{1}{n-1}} \left(\frac{(n-1)}{\kappa}(\psi + D)\right), \tag{21}$$

where D is a constant $D = \frac{\kappa}{n-1} \arcsin\left(C\rho_A^{n-1}\right) - \psi_A$.

Detail. In the case without a constraint on trajectory length, $\lambda_L = 0$. Consequently, the only one feasible root for (19) satisfying $v^* > 0$ is

$$v^* = \frac{\sqrt{1-(1-\kappa^2)C^2\rho^{2(n-1)}}}{C\kappa\rho^{n-2}}.$$

Its substitution into (18) leads to

$$\psi(\rho) = \psi_A \pm \frac{\kappa}{n-1}\arcsin\left(C\tau^{n-1}\right)\Big|_{\rho_A}^{\rho},$$

which, being rewritten as a function $\rho = \rho(\psi)$, is reduced to (21).

In the case of $n = 2$, function (21) represents an arc of a circle passing through the origin of the system of coordinates and points A and B [22]. Figure 1.3 illustrates behavior of function $\rho(\psi) = \sin^{\frac{1}{n-1}}\left(\frac{(n-1)}{\kappa}\psi\right)$ for parameters $n = 4$ and $\kappa = 0.5, 1.0, 2.0$.

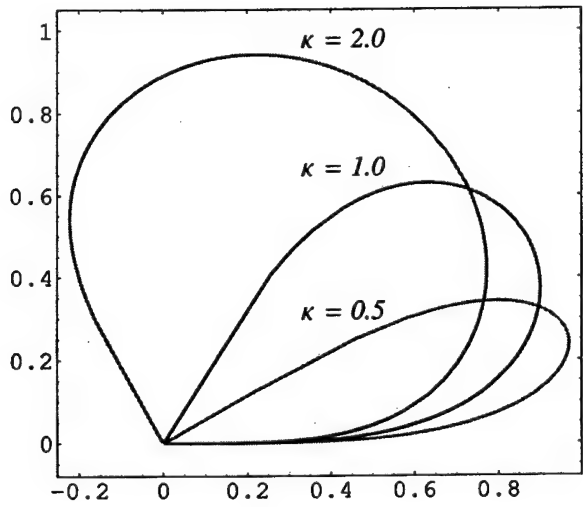

Figure 1.3. Function $\rho(\psi) = \sin^{\frac{1}{3}}\frac{3\psi}{\kappa}$.

Note if $n > 2$ constant C in (21) can be determined only when $|\psi_B - \psi_A| < \min\left\{\pi, \frac{\pi\kappa}{n-1}\right\}$, otherwise a solution to (11– 12) without constraint (13) will be unbounded.

Example 2 (the case of sphere). The second case corresponds to the optimization problem when an aircraft is modeled by a sphere, in this case $\kappa = 1$

and an optimal trajectory is presented by the explicit quadrature

$$\psi(\rho) = \psi_A \pm C \int_{\rho_A}^{\rho} \frac{\tau^{n-2} d\tau}{\sqrt{(\lambda_L \tau^n + 1)^2 - C^2 \tau^{2(n-1)}}}. \tag{22}$$

In the case of $n = 2$, quadrature (22) is reduced to the elliptic sine [22].

Detail. In the case of $\kappa = 1$, the root for (19) is given by

$$v^* = \frac{\lambda_L \rho^n + 1}{C \rho^{n-2}},$$

which being substituted into the quadrature (18) reduces it to (22).

Figure 1.4 illustrates optimal trajectories for a "spherical" aircraft ($\kappa = 1$) for $n = 4$ with different constraints on trajectory length, l_*, in trajectory's plane determined by points $(x_A, y_A) = (-0.25, 0.25)$, $(x_B, y_B) = (1.75, 0.25)$ and radar position $(0, 0)$. Figure 1.5 shows the same optimal trajectories for a "spherical" aircraft ($\kappa = 1$) for $n = 4$ with the same constraints on trajectory length, l_*, in 3D space with $(x_A, y_A, z_A) = (-0.25, 0.15, 0.2)$, $(x_B, y_B, z_B) = (1.75, 0.15, 0.2)$. Similar figure for $n = 2$ can be found in [22].

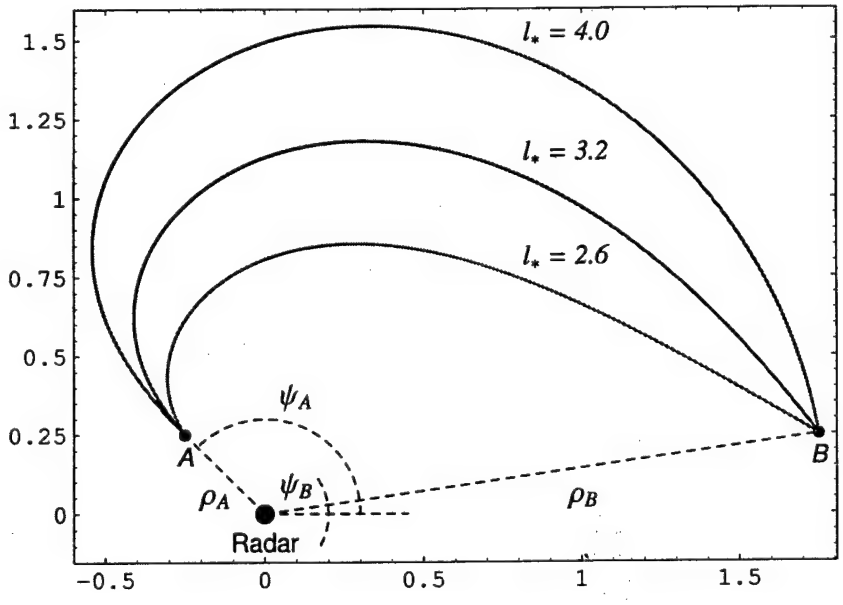

Figure 1.4. Optimal trajectories for the case of "sphere" with different constraints on the length, l_*, in the trajectories' plane.

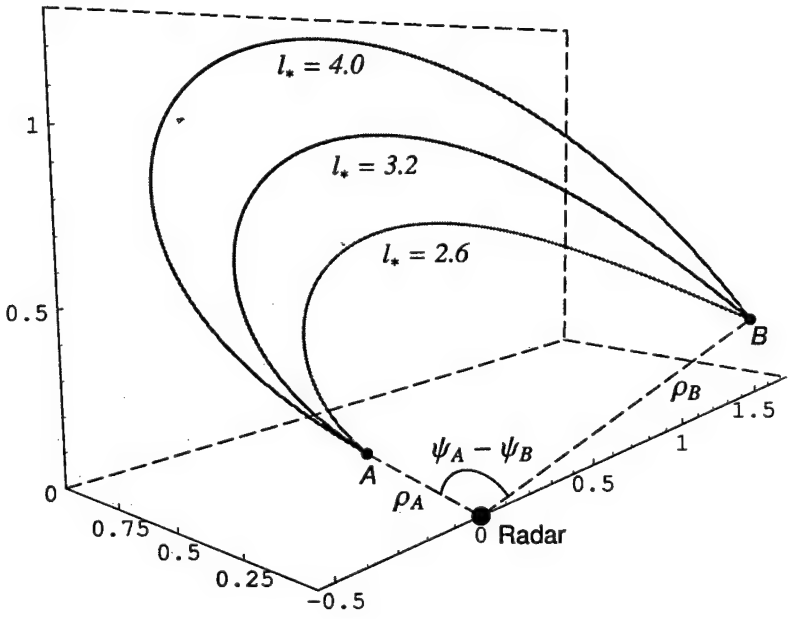

Figure 1.5. Optimal trajectories for the case of "sphere" with different constraints on the length, l_*, in 3D space.

We compared analytical solutions for optimal trajectories in the cases with radars and sensors. Analytical solution for an optimal trajectory with a constraint on trajectory length in the case of a single sensor is expressed by the elliptic sine [22]. Figure 1.6 compares optimal trajectories with the same constraint on trajectory length $l_* = 3.2$ for a "spherical" aircraft in the cases of a single sensor ($n = 2$) and single radar ($n = 4$). As expected, an optimal trajectory is more sensitive to a radar than to a sensor within proximity to an installation and vise-versa.

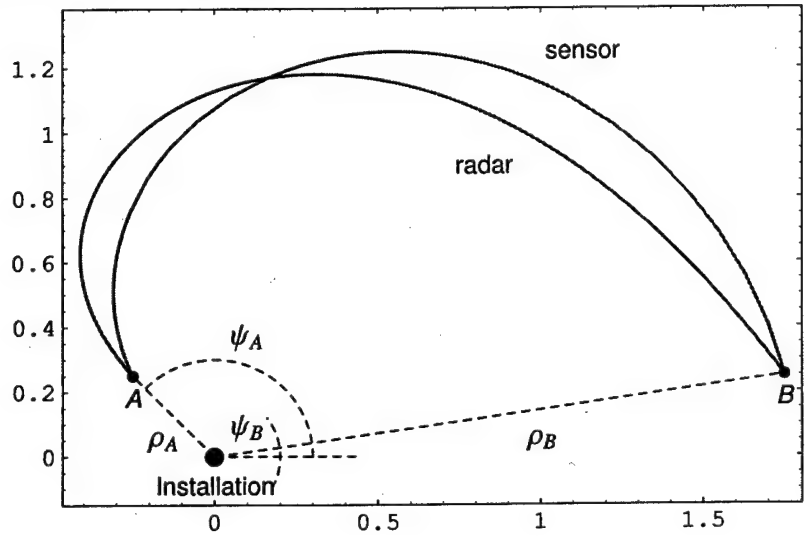

Figure 1.6. Comparison of optimal trajectories in trajectories' plane for the cases with a single sensor ($n = 2$) and single radar ($n = 4$) with the same constraint on the length, $l_* = 3.2$.

The next theorem provides some insights regarding bounds of unknowns v^* and C facilitating numerical implementation of (18) and (19).

Theorem 3.4. (estimates for λ_L, C and a feasible root for the quartic equation). *If $\lambda_L > 0$ then constant C should satisfy*

$$C \in \begin{cases} \left[0,\ n\left(\frac{\lambda_L}{n-1}\right)^{\frac{n-1}{n}}\right], \ \lambda_L \in \left[\lambda_L^-, \lambda_L^+\right], \\ \left[0,\ \min\left\{\lambda_L\rho_A + \rho_A^{-(n-1)}, \lambda_L\rho_B + \rho_B^{-(n-1)}\right\}\right], \ \lambda_L \notin \left[\lambda_L^-, \lambda_L^+\right], \end{cases} \tag{23}$$

with $\lambda_L^+ = \max\left\{\frac{n-1}{\rho_A^n}, \frac{n-1}{\rho_B^n}\right\}$, $\lambda_L^- = \min\left\{\frac{n-1}{\rho_A^n}, \frac{n-1}{\rho_B^n}\right\}$ and equation (19) has a unique root v^* within interval $[v_{\min}, v_{\max}]$ where

$$v_{\min} = \max\left\{\rho,\ C^{-1}\left(\lambda_L\rho^2 + \rho^{-(n-2)}\min\left\{1, \kappa^{-1}\right\}\right), \frac{\rho}{\kappa}\sqrt{\max\left\{C^{-2}\rho^{-2(n-1)} + \kappa^2 - 1,\ 0\right\}}\right\}, \tag{24}$$

$$v_{\max} = C^{-1}\kappa^{-1}\left(\kappa\lambda_L\rho^2 + \rho^{-(n-2)} + C\rho\sqrt{\max\{1,\kappa^2\}-1}\right). \quad (25)$$

Proof. All estimates for C and root v^* are obtained by analyzing equation (19) with respect to feasibility of v^* and the assumption of λ_L's positivity. The first part establishes bounds for C depending on λ_L. Using inequality $v = \sqrt{(\rho'_\psi)^2+\rho^2} \geq \rho$ or $v \geq \rho$, equation (19) is reduced to $\rho^{n-1}\left(Cv - \lambda_L\rho^2\right) - v \leq 0$ or, equivalently, to $\lambda_L\rho^{n+1} \geq \left(C\rho^{n-1}-1\right)v$. Applying $v \geq \rho$ again, the last inequality is reduced to $\lambda_L\rho^n \geq C\rho^{n-1}-1$, which is rearranged in the form

$$C \leq \lambda_L\rho + \rho^{-(n-1)}. \quad (26)$$

Since (26) holds for all feasible ρ, we obtain $C \leq \min_\rho\left(\lambda_L\rho + \rho^{-(n-1)}\right)$. Expression $\lambda_L\rho + \rho^{-(n-1)}$ is a convex function with respect to ρ, achieving its global minimum at $\rho_0 = \left(\frac{\lambda_L}{n-1}\right)^{-\frac{1}{n}}$. Consequently, if $\min\{\rho_A,\rho_B\} \leq \rho_0 \leq \max\{\rho_A,\rho_B\}$ (equivalently $\lambda_L \in [\lambda_L^-,\lambda_L^+]$) then ρ_0 is feasible and $C \leq \lambda_L\rho_0 + \rho_0^{-(n-1)} = n\left(\frac{\lambda_L}{n-1}\right)^{\frac{n-1}{n}}$. If $\rho_0 \notin [\min\{\rho_A,\rho_B\}, \max\{\rho_A,\rho_B\}]$ (or equivalently $\lambda_L \notin [\lambda_L^-,\lambda_L^+]$) then we are not guaranteed that in a particular example, function $\lambda_L\rho+\rho^{-(n-1)}$ will achieve its global minimum at ρ_0, since ρ_0 may not be feasible. However, in this case, at least the following weak estimate should hold $C \leq \min\left\{\lambda_L\rho_A + \rho_A^{-(n-1)},\ \lambda_L\rho_B + \rho_B^{-(n-1)}\right\}$. Based on $v \geq 0$ and the assumption of $\lambda_L \geq 0$, positivity of C is obvious. Indeed, rewriting (19) in the form

$$\rho^{n-2}\left(Cv - \lambda_L\rho^2\right)\sqrt{\kappa^2v^2 + (1-\kappa^2)\rho^2} = v \geq 0,$$

we obtain $Cv \geq \lambda_L\rho^2$, which concludes that $C > 0$. This finalizes the proof of formula (23).

The second part establishes bounds for v^*, i.e. interval $[v_{\min}, v_{\max}]$ containing a single root v^*. This part includes the following consecutive steps.

- The first lower estimate for v^* is obtained by expressing λ_L from (19) and satisfying the condition $\lambda_L \geq 0$. That is, from

$$\lambda_L = \frac{v}{\rho^2}\left(C - \frac{1}{\rho^{n-2}\sqrt{\kappa^2v^2+(1-\kappa^2)\rho^2}}\right) \geq 0,$$

we have

$$v \geq \frac{\rho}{\kappa}\sqrt{\max\left\{C^{-2}\rho^{-2(n-1)} + \kappa^2 - 1,\ 0\right\}}. \quad (27)$$

Then we utilize upper and lower estimates for $\sqrt{\kappa^2 v^2 + (1-\kappa^2)\rho^2}$ depending on whether $\kappa \leq 1$ or $\kappa > 1$.

- In the case of $\kappa \leq 1$, we have $\kappa v \leq \sqrt{\kappa^2 v^2 + (1-\kappa^2)\rho^2} \leq v$, which being applied to equation (19) reduces it to $\rho^{n-2}\kappa\left(Cv - \lambda_L \rho^2\right) \leq 1 \leq \rho^{n-2}\left(Cv - \lambda_L \rho^2\right)$. These inequalities give the upper and lower estimates for v when $\kappa \leq 1$

$$C^{-1}\left(\lambda_L \rho^2 + \rho^{-(n-2)}\right) \leq v \leq C^{-1}\left(\lambda_L \rho^2 + \kappa^{-1}\rho^{-(n-2)}\right). \quad (28)$$

  Note if $\kappa = 1$ then (28) provides an exact value for the root $v^* = C^{-1}\left(\lambda_L \rho^2 + \rho^{-(n-2)}\right)$.

- Analogously, in the case of $\kappa > 1$, we use

$$\kappa v - \rho\sqrt{\kappa^2 - 1} \leq \sqrt{\kappa^2 v^2 + (1-\kappa^2)\rho^2} \leq \kappa v,$$

  to reduce equation (19) to

$$\rho^{n-2}\left(Cv - \lambda_L \rho^2\right)\left(\kappa v - \rho\sqrt{\kappa^2 - 1}\right) \leq v \leq \rho^{n-2}\kappa v\left(Cv - \lambda_L \rho^2\right),$$

  where the left inequality is then transformed to

$$v \geq \rho^{n-2} v\left(C\kappa v - \lambda_L \kappa \rho^2 - C\rho\sqrt{\kappa^2 - 1}\right).$$

  Consequently, we obtain

$$\begin{aligned} v &\geq C^{-1}\left(\lambda_L \rho^2 + \kappa^{-1}\rho^{-(n-2)}\right), \\ v &\leq C^{-1}\left(\lambda_L \rho^2 + \kappa^{-1}\rho^{-(n-2)} + C\rho\sqrt{1-\kappa^{-2}}\right). \end{aligned} \quad (29)$$

Combining inequality $v \geq \rho$ with (27), (28) and (29) for both cases $k \leq 1$ and $\kappa > 1$, we obtain (24) and (25).

To prove that equation (19) has a single root in the interval $[v_{\min}, v_{\max}]$, we show that the function

$$f(v) = \rho^{n-2}\left(Cv - \lambda_L \rho^2\right)\sqrt{\kappa^2 v^2 + (1-\kappa^2)\rho^2} - v,$$

is monotonically increasing on $[v_{\min}, v_{\max}]$ and $f(v_{\min}) \leq 0$, $f(v_{\max}) \geq 0$. Consider

$$\frac{d}{dv} f(v) = \left(C\rho^{n-2}\sqrt{\kappa^2 v^2 + (1-\kappa^2)\rho^2} - 1\right) + \frac{\rho^{n-2}\left(Cv - \lambda_L \rho^2\right)\kappa^2 v}{\sqrt{\kappa^2 v^2 + (1-\kappa^2)\rho^2}}.$$

The first term and the nominator of the second term in the expression of $\frac{d}{dv}f(v)$ are increasing functions with respect to v. For $v \geq v_{\min}$ the first term is always nonnegative due to (27) and the nominator of the second term is always positive based on (28) and (29). Consequently, $\frac{d}{dv}f(v)$ is positive on $[v_{\min}, v_{\max}]$, which means that $f(v)$ is a monotonically increasing function.

Since $v_{\min}$ is the maximum of three values (24), we check the sign of $f(v)$ for each of them.

- The relation $f(\rho) = \rho^n \left(C - \lambda_L \rho - \rho^{-(n-1)}\right) \leq 0$ holds by virtue of (26).

- For $\hat{v}_{\min} = \frac{\rho}{\kappa}\sqrt{\max\left\{C^{-2}\rho^{-2(n-1)} + \kappa^2 - 1,\, 0\right\}}$, two cases are considered. If $C^{-2}\rho^{-2(n-1)} + \kappa^2 - 1 \leq 0$ (when $\kappa < 1$) then $\hat{v}_{\min} = 0$ and $f(\hat{v}_{\min}) = -\lambda_L \rho^{n+1}\sqrt{1-\kappa^2} < 0$. If $C^{-2}\rho^{-2(n-1)} + \kappa^2 - 1 > 0$ then $\hat{v}_{\min} = \frac{\rho}{\kappa}\sqrt{C^{-2}\rho^{-2(n-1)} + \kappa^2 - 1}$ and

$$f(\hat{v}_{\min}) = C^{-1}\left(C\hat{v}_{\min} - \lambda_L\rho^2\right) - \hat{v}_{\min} = -C^{-1}\lambda_L\rho^2 < 0.$$

- For $\tilde{v}_{\min} = C^{-1}\left(\lambda_L\rho^2 + \rho^{-(n-2)}\min\left\{1, \kappa^{-1}\right\}\right)$, based on

$$\sqrt{\kappa^2 v^2 + (1-\kappa^2)\rho^2} \leq v \min\{1, \kappa\},$$

we obtain

$$\begin{aligned} f(\tilde{v}_{\min}) &\leq \tilde{v}_{\min}\left(\rho^{n-2}\left(C\tilde{v}_{\min} - \lambda_L\rho^2\right)\min\{1,\kappa\} - 1\right) \\ &= \tilde{v}_{\min}\left(\min\left\{1,\kappa^{-1}\right\}\min\{1,\kappa\} - 1\right) = 0. \end{aligned}$$

Thus we established that $f(v_{\min}) \leq 0$ for $v_{\min} = \max\{\rho, \hat{v}_{\min}, \tilde{v}_{\min}\}$.

In the case of $v_{\max}$ given by (24), we use

$$\sqrt{\kappa^2 v^2 + (1-\kappa^2)\rho^2} \geq \kappa v - \rho\sqrt{\max\{1, \kappa^2\} - 1},$$

to show that

$$\begin{aligned} f(v_{\max}) &\geq \rho^{n-2}\left(Cv_{\max} - \lambda_L\rho^2\right)\left(\kappa v_{\max} - \rho\sqrt{\max\{1,\kappa^2\} - 1}\right) - v_{\max} \\ &\geq \rho^{n-2}v_{\max}\left(C\,\kappa\,v_{\max} - C\rho\sqrt{\max\{1,\kappa^2\} - 1} - \kappa\lambda_L\,\rho^2\right) - v_{\max} = 0. \end{aligned}$$

Consequently, we proved that $f(v_{\min}) \leq 0$ and $f(v_{\max}) \geq 0$, which along with the condition of $f(v)$'s monotonicity on $[v_{\min}, v_{\max}]$ guarantee existence of only a single root for $f(v)$ on $[v_{\min}, v_{\max}]$. □

Example 3 (elongated and compressed ellipsoids in the case of $n = 4$). Coordinates of points A and B are the same in all examples. In trajectory's plane $(x_A, y_A) = (-0.25, 0.25)$, $(x_B, y_B) = (1.75, 0.25)$, and in 3D space $(x_A, y_A, z_A) = (-0.25, 0.15, 0.2)$, $(x_B, y_B, z_B) = (1.75, 0.15, 0.2)$. Figures 1.7, 1.8 and 1.9 compare optimal trajectories for sphere, elongated and compressed ellipsoids. Table 1.1 presents values for the optimal risk, λ_L, C and C's estimate (23) for all considered numerical examples with different κ and l_*.

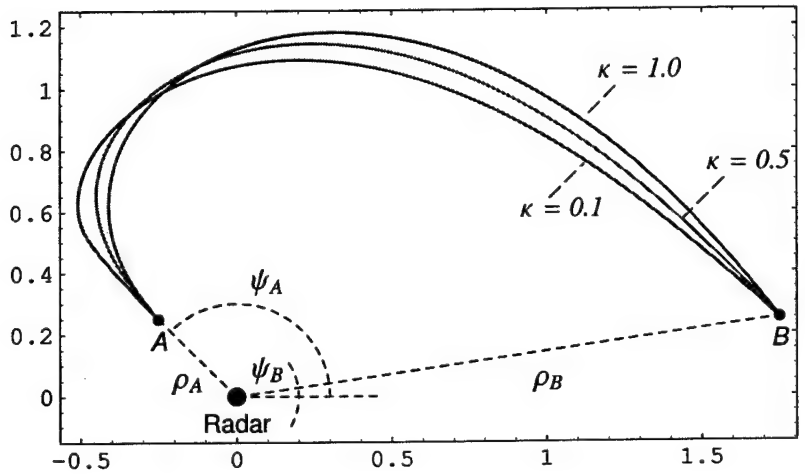

Figure 1.7. Optimal trajectories for sphere ($\kappa = 1.0$) and elongated ellipsoids ($\kappa = 0.5, 0.1$) for $n = 4$ and $l_* = 3.2$ shown in trajectory's plane.

Table 1.1. Results of numerical experiments: values of optimal risk, λ_L, C and C's estimate.

κ	l_*	*Risk*	λ_L	C	$4(\lambda_L/3)^{3/4}$
1.0	2.6	9.792116	4.763580322	5.369280470	5.658081121
1.0	3.2	8.421726	1.040107422	1.759684438	1.807289381
1.0	4.0	7.966210	0.300707031	0.712062456	0.712568693
0.1	3.2	0.468371	1.451660156	2.282777298	2.320693404
0.5	3.2	3.980716	1.993432500	2.908499104	2.943880665
2.0	3.2	12.464087	0.610351562	1.143055224	1.211725076
10.0	3.2	14.845494	0.109076172	0.251462743	0.333055390

Analyzing optimal trajectories in Figures 1.6–1.9 and computational results in Table 1.1 we conclude the following

- *The optimal risk is more sensitive to variation of the shape of ellipsoid (parameter κ), than to the variation of a trajectory's total length, l_*.*

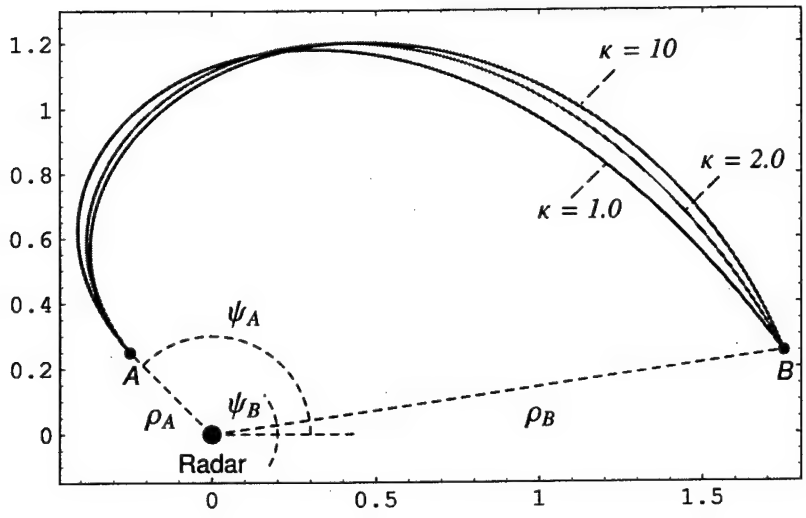

Figure 1.8. Optimal trajectories for sphere ($\kappa = 1.0$) and compressed ellipsoids ($\kappa = 2.0$, 10) for $n = 4$ and $l_* = 3.2$ shown in trajectory's plane.

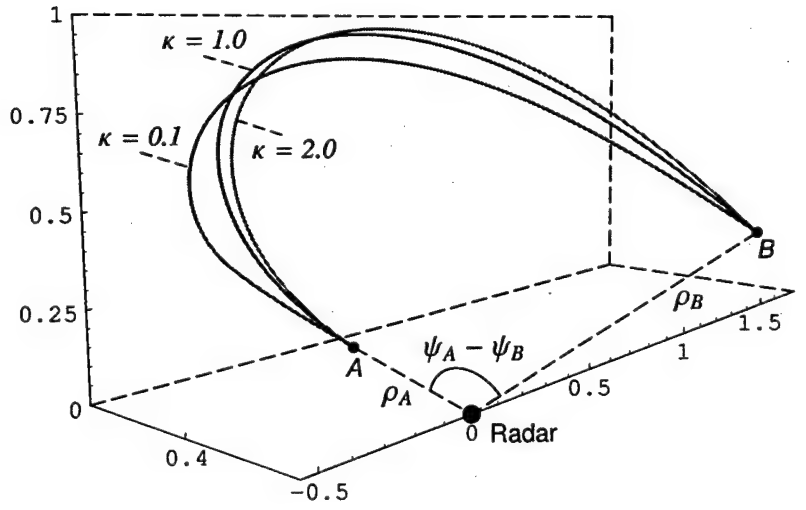

Figure 1.9. Optimal trajectories for sphere ($\kappa = 1.0$), elongated ($\kappa = 0.1$) and compressed ($\kappa = 2.0$) ellipsoids for $n = 4$ and $l_* = 3.2$ in 3D space.

- *Optimal trajectories for different κ (especially for $\kappa > 1$) are close to each other, which indicates that a variation of ellipsoid shape has no strong effect on the geometry of an optimal trajectory.*
- *Within proximity to an installation, an optimal trajectory is more sensitive to a radar-installation than to a sensor-installation and in the area remote from the installation the effect is opposite.*

4. Network Flow Optimization Approach

The calculus of variations approach reduces the optimization problem to the vectorial nonlinear differential equation. Obtaining an analytical solution to this equation in the case of arbitrary number of detecting installations is still an open issue. Certainly, various gradient-based techniques may solve the equation numerically. However, regardless of efficiency of those techniques (although, this issue is also questionable due to strong nonlinearity of the equation), most of them provide only locally optimal solution. This section develops a discrete optimization approach generating globally optimal trajectories.

We propose network flow (NF) optimization approach to directly solve the original problem. This approach reduces optimal risk path generation with a constraint on the length to the Constrained Shortest Path Problem (CSPP) for a 3D network, which can efficiently be solved by NF optimization algorithms. There are several advantages of using NF optimization

- *Among all feasible approximated trajectories in a considered network, it finds a globally optimal one.*
- *Its complexity (running time) depends neither on a number of installations in a network nor on power n in the risk functional (2).*
- *It can readily be generalized for the case with an actual-tabulated radar cross-section (RCS) (i.e. when RCS is not a smooth function).*

However, due to NP-hard nature of the CSPP, no polynomial algorithm solves the CSPP exactly. It means that in a worst case, computational time for the CSPP will exponentially depend on the number of arcs in a network. Consequently, coarseness of the network should be specified reasonably.

4.1. Network Structure

We assume an admissible deviation domain for aircraft trajectory to be an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N} = \{1, \ldots, n\}$ is the set consisting of n nodes and $\mathcal{A}$ is the set of undirected arcs. A trajectory $(x(.), y(.), z(.))$ is approximated by a path $\mathcal{P}$ in the graph $\mathcal{G}$, where path $\mathcal{P}$ is defined as a sequence

of nodes $\langle j_0, j_1, \ldots, j_p \rangle$ such that $j_0 = A$, $j_p = B$ and $\langle j_{k-1}, j_k \rangle \in \mathcal{A}$ for all k from 1 to p. Let graph $\mathcal{G}$ be a 3D grid of nodes (rectangular parallelepiped) of $n_x \times n_y \times n_z$ size with edges oriented along coordinate axes x, y, z and having n_x, n_y and n_z numbers of unit segments in each edge, respectively. Similarly, in 2D case graph $\mathcal{G}$ is a 2D grid of nodes (rectangle) of $n_x \times n_y$ size with edges oriented along coordinate axes x, y and having n_x and n_y numbers of unit segments in each edge, respectively. Structures of arcs assigned in $\mathcal{G}$ in 2D and 3D cases are shown in Figures 1.10 and 1.11, respectively. A 2D network with arcs structure, shown in Figure 1.10, contains $(n_x + 1)(n_y + 1)$ nodes and

$$2(8\, n_x n_y - n_x - n_y)$$

arcs, where $n_x \geq 1$ and $n_y \geq 1$. In 3D case the total number of nodes and arcs in an undirected $\mathcal{G}$ with arcs structure, as shown in Figure 1.11, are $(n_x + 1)(n_y + 1)(n_z + 1)$ and

$$2(49\, n_x n_y n_z - 8\, n_x n_y - 8\, n_x n_z - 8\, n_y n_z + n_x + n_y + n_z),$$

for $n_x \geq 1$, $n_y \geq 1$, and $n_z \geq 1$. For instance, the case of $n_x = 1$, $n_y = 1$, and $n_z = 1$ corresponds to a single cube with 12 liner arcs, 12 planar arcs and 4 3D arcs (see Figure 1.11). All these numbers should be doubled due to "undirectness" of the graph. Thus, the total number of arcs just in a single cube is 56. Moreover, in order to provide sufficient amount of feasible directions for a trajectory (i.e. to avoid "naive discretization," sometimes referred to as the digitization bias, [18]), we assign not only axis and diagonal but also so-called "long-diagonal" arcs connecting opposite vertexes of any two neighbor cubes (see Figure 1.11). However, in this case, network structure becomes very condense. For example, a relatively small 3D undirected network of $40 \times 40 \times 40$ contains about 69,000 nodes and 6,200,000 arcs. It would be naive to assume that even a very efficient NF algorithm is capable to find a constrained shortest path in this network within seconds (at least at the current moment of the technological progress). Obviously, a main task in this case is development of efficient network structures with relatively small numbers of arcs and nodes, while preserving flexibility for trajectory approximation, rather than finding the most efficient NF algorithm (although this is also a quite legitimate question). However, the chapter only partially addresses the issue of efficient network structures, since this is a separate subject for discussion, which calls for a separate publication. Zabarankin et al. [22] showed that existing NF algorithms [7] are quite efficient in finding a constrained shortest path in a network with about 100,000 arcs.

Smoothing procedure and curvature constraint. Number of arcs can significantly be reduced by smoothing network structure. Starting from both $j_0 = A$ and $j_p = B$ nodes along all directions outgoing from j_0 and j_p, we retain only

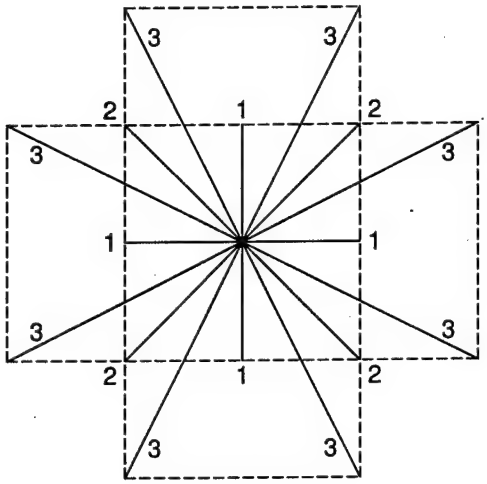

Figure 1.10. Structure of arcs in every node in a 2D network: "1" – axis arcs, "2" – diagonal arcs, "3" – long diagonal arcs.

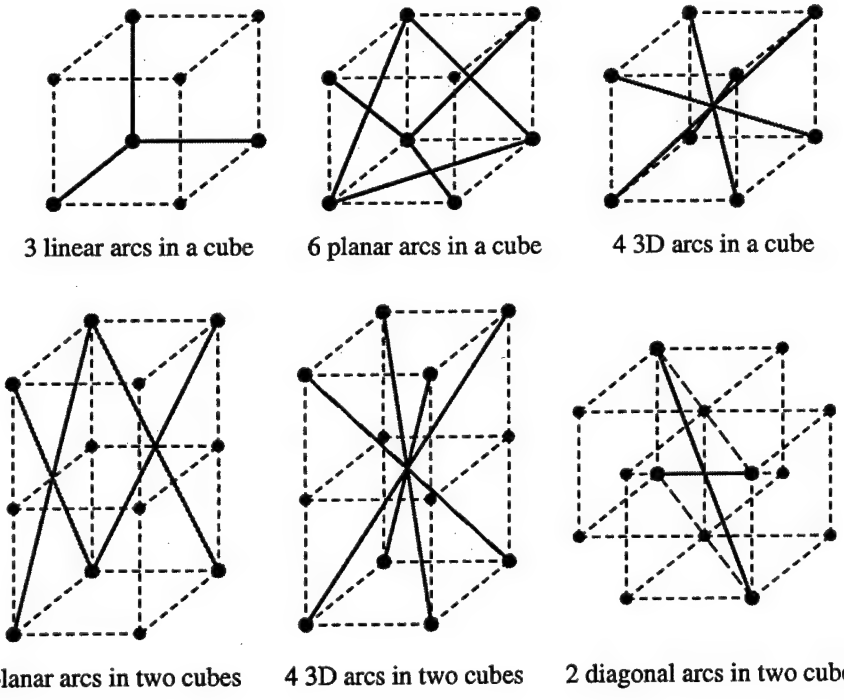

Figure 1.11. Structure of arcs in every node in a 3D network.

those pairs of arcs, which produce the angle not greater than, for instance, $\frac{\pi}{6}$, see Figure 1.12.

Let $\mathcal{N}_i$ be the set of nodes connected to node i. For instance, for 2D network structure shown in Figure 1.10, set $\mathcal{N}_i$ for each i consists of 16 nodes producing 4 axis, 4 diagonal and 8 long diagonal arcs with node i. Let $\mathcal{T}_i^A$ be a subgraph starting from node $j_0 = A$ along arc $\langle j_0, i\rangle$, $i \in \mathcal{N}_{j_0}$. Now $\mathcal{T}_i^A = \{j_0, i\}$. If $j \in \mathcal{N}_i/\{j_0\}$ satisfies condition $\mathbf{e}_{j_0 i} \cdot \mathbf{e}_{ij} \geq \frac{\sqrt{3}}{2}$, where $\mathbf{e}_{ij}$ is the unit vector along arc $\langle i, j\rangle$, then j is added to the subgraph $\mathcal{T}_i^A = \mathcal{T}_i^A \cup \{j\}$. The next step is to examine new added nodes, i.e. for all $j \in \mathcal{T}_i^A/\{j_0, i\}$ check $\mathbf{e}_{ij} \cdot \mathbf{e}_{jk} \geq \frac{\sqrt{3}}{2}$, $k \in \mathcal{N}_j$. If node k satisfies this condition then $\mathcal{T}_i^A = \mathcal{T}_i^A \cup \{k\}$ and so on. $\mathcal{T}_i^A$ is full when there is no nodes left satisfying the condition. Similarly, we construct $\mathcal{T}_i^A$ for all $i \in \mathcal{N}_{j_0}$. Then the whole process is repeated for $\mathcal{T}_i^B$.

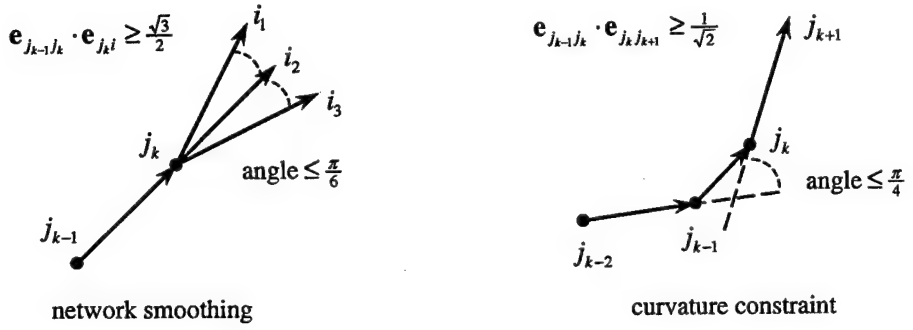

Figure 1.12. Network smoothing and curvature constraint.

To avoid sharp turns in aircraft trajectory, we may use a curvature constraint, which also can be imposed by aircraft control limitations. Analytically, for any given triad of arcs, curvature constraint is expressed by $\mathbf{e}_{j_{k-2}j_{k-1}} \cdot \mathbf{e}_{j_k j_{k+1}} \geq \cos\alpha$, where α, for instance, may be $\frac{\pi}{4}$. In general, α should be a function of the length of the middle arc $\langle j_{k-1}, j_k\rangle$, since a constant constraint on trajectory curvature may prevent from obtaining an optimal solution.

Finding a globally optimal risk path subject to length constraint is the task for NF optimization. Network structure smoothing will be integrated into a NF algorithm as a condition eliminating inadmissible arcs in a network rather than implemented as a separate procedure. Consequently, smoothing condition is considered now as an adjustment for the NF algorithm rather than property of network structure.

4.2. Approximation Scheme

Several schemes for approximation of optimization problem (5) are available. We consider one of them. Let vector $\mathbf{r}_{j_k}$ with components $x(j_k)$, $y(j_k)$ and $z(j_k)$ determine position of node j_k. Then a path $\mathcal{P} = \left\{\mathbf{r}_{j_1}, \mathbf{r}_{j_2}, \ldots, \mathbf{r}_{j_p}\right\}$ is a piece-wise linear curve (broken line) with vertexes at points $\mathbf{r}_{j_k}$, $k = \overline{1,p}$. Any point on the arc $\langle j_{k-1}, j_k\rangle$ can be defined by vector $\mathbf{r}_k(t) = (1-t)\,\mathbf{r}_{j_{k-1}} + t\,\mathbf{r}_{j_k}$ with $t \in [0,1]$. Thus, length differential ds and derivative $\dot{\mathbf{r}}$ for each arc are

$$ds = ||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||\,dt, \qquad \dot{\mathbf{r}}_k = \frac{\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}}{||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||}, \qquad k = \overline{1,p}.$$

Using approximations for $\mathbf{r}$, $\dot{\mathbf{r}}$ and ds, functional (4) and trajectory length are presented, respectively

$$\mathcal{F}(\mathbf{r}, \dot{\mathbf{r}}) \approx \sum_{k=1}^{p} ||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}|| \int_0^1 L\left(\mathbf{r}_k(t), \dot{\mathbf{r}}_k\right) dt = \sum_{k=1}^{p} \mathcal{C}(\mathbf{r}_{j_{k-1}}, \mathbf{r}_{j_k}), \quad (30)$$

$$l \approx \sum_{k=1}^{p} ||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||, \quad (31)$$

where $||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||$ and $\mathcal{C}(\mathbf{r}_{j_{k-1}}, \mathbf{r}_{j_k})$ are the length and risk index of the arc $\langle j_{k-1}, j_k\rangle$, respectively. To derive the formula for $\mathcal{C}(\mathbf{r}_{j_{k-1}}, \mathbf{r}_{j_k})$ we compute the risk accumulated along the arc $\langle j_{k-1}, j_k\rangle$ from the i^{th} radar located at $\mathbf{q}_i = (a_i, b_i, c_i)$. Substituting $\mathbf{r}_{ik}(t) = \mathbf{r}_k(t) - \mathbf{q}_i$ into (2), we have

$$\begin{aligned}
&\mathcal{C}(\mathbf{r}_{j_{k-1}}, \mathbf{r}_{j_k}) \\
&= \frac{2\kappa}{1+\kappa^2} \sum_{i=1}^{N} w_i \int_0^1 \frac{\sqrt{||\mathbf{r}_{ik}(t)||^2\,||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||^2 + (\kappa^2-1)\;\mathbf{r}_{ik}(t)\cdot(\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}})^2}}{||\mathbf{r}_{ik}(t)||^{n+1}}\,dt \\
&= \frac{2\kappa}{1+\kappa^2} \sum_{i=1}^{N} w_i \int_0^1 \frac{\sqrt{l_{i,j_{k-1}}^2 \sin^2\phi_{i,j_{k-1}j_k} + \kappa^2\;l_{i,j_{k-1}}\cos\phi_{i,j_{k-1}j_k} + t\,\Delta s_{j_{k-1}\,j_k}{}^2}\,dt}{\left(l_{i,j_{k-1}}^2 \sin^2\phi_{i,j_{k-1}j_k} + l_{i,j_{k-1}}\cos\phi_{i,j_{k-1}j_k} + t\,\Delta s_{j_{k-1}\,j_k}{}^2\right)^{\frac{n+1}{2}}},
\end{aligned} \quad (32)$$

where

$$l_{i,j_k} = ||\mathbf{r}_{j_k} - \mathbf{q}_i||, \qquad \Delta s_{j_{k-1}\,j_k} = ||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||,$$

and $\phi_{i,\,j_{k-1}j_k} \in [0,\pi]$ is the angle between vectors $\mathbf{r}_{j_{k-1}} - \mathbf{q}_i$ and $\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}$ (see Figure 1.13), i.e.

$$\phi_{i,\, j_{k-1}j_k} = \arccos\left(\frac{(\mathbf{r}_{j_{k-1}} - \mathbf{q}_i)\cdot(\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}})}{||\mathbf{r}_{j_{k-1}} - \mathbf{q}_i||\;||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||}\right).$$

Figure 1.13 illustrates a 3D network for solving the risk minimization problem. Broken line AB is a path in the area with the i^{th} radar, while $\Delta s_{j_{k-1}\, j_k}$ is the length of arc $\langle j_{k-1}, j_k\rangle$ between nodes j_{k-1} and j_k in this path. Magnitude $\phi_{i,\, j_{k-1}j_k}$ is the angle between vector $\mathbf{r}_{j_{k-1}} - \mathbf{q}_i$ and arc $\langle j_{k-1}, j_k\rangle$ directed from node j_{k-1} to node j_k.

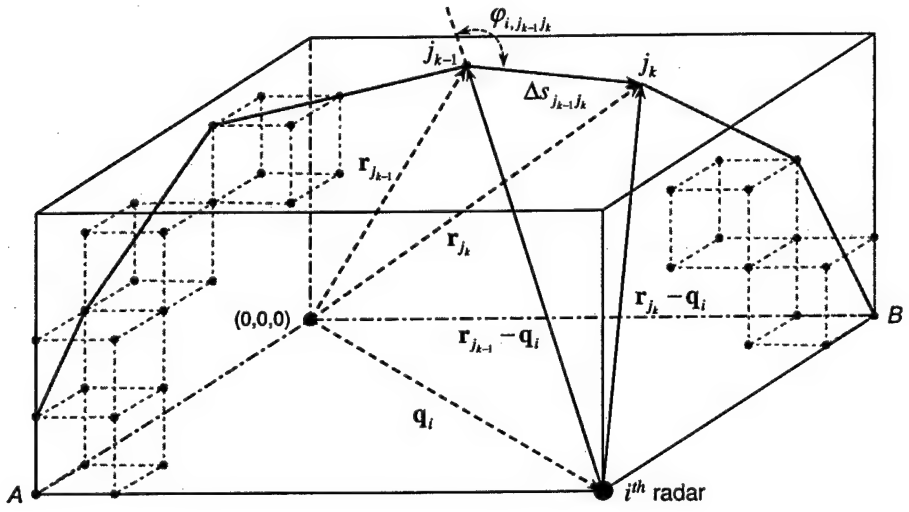

Figure 1.13. 3D network for solving the risk minimization problem – broken line AB is a path of the aircraft.

Integral (32) can efficiently be approximated by the Gaussian quadrature. If $f(t)$ is a bounded smooth function on $[0, 1]$ then the Gaussian quadrature is

$$\int_0^1 f(t)\,dt \approx \sum_{j=1}^{J} h_j\, f(t_j),$$

where h_j and $t_j \in [0, 1]$ are known for any given J. For instance, Table 1.2 presents values t_j and weight coefficients h_j for the Gaussian quadrature for $J = 16$.

Consequently, using a direct method of Calculus of Variations, problem (5) is approximated by

Table 1.2. Values t_j and weight coefficients h_j for the Gaussian quadrature for $J = 16$.

j	t_j	h_j
1	0.048307665687738316235	0.096540088514727800567
2	0.14447196158279649348 5	0.095638720079274859419
3	0.23928736225213707454 5	0.093844399080804565639
4	0.33186860228212764978 0	0.091173878695763884713
5	0.42135127613063534536 4	0.087652093004403811143
6	0.50689908932229390024	0.083311924226946755222
7	0.58771575724076232904 1	0.078193895787070306472
8	0.66304426693021520097 5	0.072345794108848506225
9	0.73218211874028968038 7	0.065822222776361846838
10	0.79448379596794240696 3	0.058684093478535547145
11	0.84936761373256997013 4	0.050998059263376176196
12	0.89632115576605212396 5	0.042835898022226680657
13	0.93490607593773968917 1	0.034273862913021433103
14	0.96476225558750643077 4	0.025392065309262059456
15	0.98561151154526833540 0	0.016274394730905670605
16	0.99726386184948156354 5	0.007018610009470096600

$$\begin{aligned} \min_{P} \quad & \sum_{k=1}^{p} \mathcal{C}(\mathbf{r}_{j_{k-1}}, \mathbf{r}_{j_k}) \\ s.\,t. \quad & \sum_{k=1}^{p} ||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}|| \le l_*, \\ & \mathbf{r}_{j_0} = \mathbf{r}_A, \;\; \mathbf{r}_{j_p} = \mathbf{r}_B. \end{aligned} \tag{33}$$

If for all $k = \overline{1,p}$, $\mathbf{r}_{j_k}$ is variable (not fixed in nodes of a network), then, in the case of active constraint $\sum_{k=1}^{p} ||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}|| = l_*$, optimality conditions for (33) can be derived by standard calculus

$$\frac{\partial}{\partial \mathbf{r}_{j_k}} \left(\mathcal{C}(\mathbf{r}_{j_{k-1}}, \mathbf{r}_{j_k}) + \mathcal{C}(\mathbf{r}_{j_k}, \mathbf{r}_{j_{k+1}}) \right) = -\gamma \left(\mathbf{e}_{j_{k-1}\, j_k} - \mathbf{e}_{j_k\, j_{k+1}} \right), \tag{34}$$

where $k = \overline{1,(p-1)}$, γ is the Lagrange multiplier for the constraint in (33) and $\mathbf{e}_{j_{k-1}\, j_k} = \frac{\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}}{||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||}$. System (34) may be solved numerically by a gradient-based algorithm. However, in this case, we most likely obtain only locally optimal solution. Moreover, instead of solving (34) we could numerically solve differential equation (7).

4.3. Reduction to the Constrained Shortest Path Problem

To formulate (33) as a network flow optimization problem, let

$$c_{j_{k-1}\,j_k} = C(\mathbf{r}_{j_{k-1}}, \mathbf{r}_{j_k}), \qquad \Delta s_{j_{k-1}\,j_k} = ||\mathbf{r}_{j_k} - \mathbf{r}_{j_{k-1}}||,$$

and values $R(\mathcal{P})$ and $l(\mathcal{P})$ define the total cost (risk) and weight (length) accumulated along the path $\mathcal{P}$, respectively,

$$R(\mathcal{P}) = \sum_{k=1}^{p} c_{j_{k-1}\,j_k}, \qquad l(\mathcal{P}) = \sum_{k=1}^{p} \Delta s_{j_{k-1}\,j_k}.$$

Thus, each arc $\langle j_{k-1}, j_k \rangle \in \mathcal{A}$ is associated with its length $\Delta s_{j_{k-1}\,j_k}$ and nonnegative cost $c_{j_{k-1}\,j_k}$. The path $\mathcal{P}$ is weight feasible if the total weight $l(\mathcal{P})$ is at most l_*, i.e. $l(\mathcal{P}) \leq l_*$. Consequently, the CSPP is finding such a feasible path $\mathcal{P}$ from point A to point B that minimizes cost $R(\mathcal{P})$

$$\begin{aligned} \min_{\mathcal{P}} \quad & \sum_{k=1}^{p} c_{j_{k-1}\,j_k} \\ s.\,t. \quad & \sum_{k=1}^{p} \Delta s_{j_{k-1}\,j_k} \leq l_*. \end{aligned} \tag{35}$$

The difference between (33) and (35) is that (33) still preserves analytical properties of the risk and length, whereas (35) completely "forgets" about the nature of obtained $c_{j_{k-1}\,j_k}$ and $\Delta s_{j_{k-1}\,j_k}$ values. The CSPP (35) is closely related to the Shortest Path Problem with Time Windows (SPPTW) and also to the Resource Constrained Shortest Path Problem (RCSPP), which uses a vector of weights, or resources, rather than a scalar. These problems are solved in column generation approaches for Vehicle Routing Problems with Time Windows (VRPTW) and in long-haul aircraft routing problems. Under the assumption of cost and weight integrality, the CSPP was shown to be a *NP-hard* problem [8]. It means that in a worse case, the CSPP is solved in time exponentially depending on the number of arcs. Algorithms for solving the CSPP are divided into three major categories:

- *Label-setting algorithms based on dynamic programming methods*
- *Scaling algorithms*
- *Algorithms based on the Lagrangean relaxation approach*

The label setting algorithm is the most efficient in the case when the weights are positive [6]. The Lagrangean relaxation algorithm is based on the subgradient optimization [2] and cutting plane [10] methods, and efficient for solving

the Lagrangean dual problem of the CSPP in the case of one resource. Scaling algorithms use two fully polynomial approximation schemes for the CSPP based on cost scaling and rounding [11]. The first scheme is a geometric bisection search whereas the second one iteratively extends paths. We solve the CSPP (35) by the Label Setting Algorithm (LSA) with a preprocessing procedure [8].

4.4. The Label Settings Algorithm with Preprocessing Procedure

The Preprocessing Procedure and Label Setting Algorithm (LSA) are two consecutive stages in finding a constrained shortest path. The objective of preprocessing is to reduce the original graph by eliminating all arcs and nodes such that any path containing them is infeasible or does not improve current cost upper bound. To discuss the algorithm in detail, let us denote the arc's nodes j_{k-1} and j_k by i and j, respectively. For each node i, we consider the path obtained by appending the least cost path from the source node s to i to the least cost path from i to the sink node t. If the total cost accumulated along the new path is at least the current cost upper bound, then the use of node i cannot improve a known feasible solution. Hence, node i and all arcs incident to it can be deleted from the graph. If the total cost is less than the upper bound and the path is feasible, then the upper bound can be updated and the process continues with the improved upper bound. Similar, for each arc $\langle i, j \rangle$, we consider the path obtained by appending the least cost path from s to i to the least cost path from j to t, via arc $\langle i, j \rangle$. If the total cost accumulated along the new path is at least equal to the current cost upper bound, then we can delete arc $\langle i, j \rangle$ from the graph. If the total cost is less than the upper bound and the path is feasible then the upper bound can be updated. The preprocessing procedure is presented in the pseudo-code form below.

Preprocessing Algorithm for the CSPP

Step 0: Let $U = C(n-1)$ where $C = \max_{\langle i,j \rangle \in \mathcal{A}} c_{ij}$.

Step 1: Find the *minimum cost* paths from source node $s = A$ with arc costs given by c_{ij}. Let $\mathcal{Q}^c_{sj}$ be the least cost path from s to j and α^c_j be the cost of the path: $\alpha^c_j = R(\mathcal{Q}^c_{sj})$.
If there is no path from s to the sink node $t = B$ **then** stop; the problem is infeasible.
If $l(\mathcal{Q}^c_{st}) \leq l_*$ **then** $\mathcal{Q}^c_{st}$ is the optimal path.

Step 2: Find the *minimum cost* paths from all nodes to t with arc costs given by c_{ij}. Let $\mathcal{Q}^c_{jt}$ be the least cost path from j to t and β^c_j be the

cost of the path: $\beta_j^c = R(\mathcal{Q}_{jt}^c)$.

Step 3: Find the *minimum length* paths from s to all nodes with arc lengths given by Δs_{ij}. $\mathcal{Q}_{sj}^l$ is the minimum length path from s to j and α_j^l is the length of this path: $\alpha_j^l = l(\mathcal{Q}_{sj}^l)$.
If $l(\mathcal{Q}_{st}^l) > l_*$ **then** stop; the problem is infeasible.
If $l(\mathcal{Q}_{st}^l) \leq l_*$ and $R(\mathcal{Q}_{st}^l) < U$ **then** set $U = R(\mathcal{Q}_{st}^l)$.

Step 4: Find the *minimum length* paths from all nodes to t with the arc lengths given by Δs_{ij}. $\mathcal{Q}_{jt}^l$ is the least length path from j to t and β_j^l is the length of this path: $\beta_j^l = l(\mathcal{Q}_{jt}^l)$.

Step 5: **For** all $j \in V \backslash \{s, t\}$ **do**
if $\alpha_j^l + \beta_j^l > l_*$ **then** delete node j and all arcs incident to it;
if $\alpha_j^c + \beta_j^c \geq U$ **then** delete node j and all arcs incident to it;
end

Step 6: **For** all $\langle i, j \rangle \in \mathcal{A}$ **do**
if $\alpha_i^l + \Delta s_{ij} + \beta_j^l > l_*$ **then** delete $\langle i, j \rangle$
else if $\alpha_i^c + c_{ij} + \beta_j^c \geq U$ **then** delete $\langle i, j \rangle$
else if $l(\mathcal{Q}_{si}^c) + \Delta s_{ij} + l(\mathcal{Q}_{jt}^c) \leq l_*$ **then** $U = \alpha_i^c + c_{ij} + \beta_j^c$;
end

Step 7: **If** during steps 5 and 6 the graph changed **then** **goto** Step 1,
else set $L = \alpha_t^c$ and stop.

End.

The next stage after the preprocessing procedure is the Label Setting Algorithm. The idea of the algorithm is to use a set of labels for each node and compare the labels to one another. Each label on a node represents a different path from node s to that node and consists of a pair of numbers representing the cost and weight of the corresponding path. No labels having the same cost are stored and for each label on a node, any other label on that node with a lower cost must have a greater weight. Let I_i be the index set of labels on node i and for each $k \in I_i$ let $\mathcal{P}_i^k$ denote a path from s to i with weight W_i^k and cost C_i^k. Pair (W_i^k, C_i^k) is the label of node i and $\mathcal{P}_i^k$ is the path corresponding to it. For two labels (W_i^k, C_i^k) and (W_i^q, C_i^q), corresponding to two different paths $\mathcal{P}_i^k$ and $\mathcal{P}_i^q$, respectively, (W_i^k, C_i^k) dominates (W_i^q, C_i^q) if $W_i^k \leq W_i^q$, $C_i^k \leq C_i^q$, and the labels are not equal. Label (W_i^k, C_i^k) is efficient if it is not dominated by any other label at node i, i. e. if $(l(\mathcal{P}), R(\mathcal{P}))$ does not dominate (W_i^k, C_i^k) for all paths $\mathcal{P}$ from s to i. A path is efficient if the label it corresponds to is

efficient. The LSA finds all efficient labels in every node. Starting without any labels on any node, except for label $(0,\ 0)$ on node s, the algorithm extends the set of all labels by treating an existing label on a node, that is, by extending the corresponding path along all outgoing arcs. Let L_i be the set of labels on node i and let $T_i \subseteq I_i$ index the labels on node i, which have been treated. The algorithm proceeds until all labels have been treated, i.e. until $I_i \backslash T_i = \emptyset$ for all $i \in V \backslash \{t\}$.

The Label Setting Algorithm (LSA) with smoothing condition

Step 0: *Initialization*

Run Preprocessing Algorithm for the CSPP to find U, β_j^c, β_j^l and $\mathcal{Q}_{jt}^c \quad \forall j \in V \backslash \{t\}$.
Set $L_s = \{(0,\ 0)\}$ and $L_i = \emptyset$ for all $i \in V \backslash \{s\}$.
Initialize I_i accordingly for each $i \in V$.
Set $T_i = \emptyset$ for each $i \in V$.

Step 1: *Selection of the label to be treated*

If $\bigcup_{i \in V} (I_i \backslash T_i) = \emptyset$ **then** stop; all efficient labels have been generated.
Else choose $i \in V$ and $k \in I_i \backslash T_i$ so that W_i^k is minimal.

Step 2: *Treatment of label* (W_i^k, C_i^k)

For all $\langle i, j \rangle \in \mathcal{A}$ **do**
If $\left(\mathbf{e}_{(\leftarrow i)i} \cdot \mathbf{e}_{ij} \geq \epsilon\right)$ /**smoothing condition:* $\epsilon = \frac{\sqrt{3}}{2}$; $(\leftarrow i)$ is a predecessor node */
If $(W_i^k + \Delta s_{ij} + \beta_j^l \leq l_*)$
If $(C_i^k + c_{ij} + \beta_j^c < U)$
If $(W_i^k + \Delta s_{ij}, C_i^k + c_{ij})$ is not dominated by $(W_j^q, C_j^q) \quad \forall q \in I_j$
then set $L_j = L_j \cup \{(W_i^k + \Delta s_{ij}, C_i^k + c_{ij})\}$ and update I_j
If $(W_i^k + \Delta s_{ij} + l(\mathcal{Q}_{jt}^c) \leq l_*)$ **then** $U = C_i^k + c_{ij} + \beta_j^c$.
end

Step 3: Set $T_i = T_i \cup \{k\}$, **goto** Step 1.

End.

5. Numerical Experiments

Zabarankin et al. [22] demonstrated efficiency of the NF optimization approach in optimal trajectory generation in 2D space with arbitrary number of sensors ($n = 2$) for the case of sphere ($\kappa = 1$). This section tests complexity of the CSPP (35) and flexibility of the proposed 3D network structure in optimal trajectory generation with variable aircraft RCS in 3D space. We computed discrete solutions using 2D and 3D networks for the same data considered in Examples 2 and 3 in the case of a single radar ($n = 4$). Radar position, coordinates of points A and B and testing values for a constraint on trajectory length are exactly the same. In 3D case, the CSPP was solved by the LSA with and without smoothing condition. All calculations were conducted using a PC with *Xeon 3.08 GHz* and *3.37 Gb of RAM*.

We used 2D and 3D networks, with structures as shown in Figures 1.10 and 1.11, to compare discrete optimization trajectories with analytical ones in the case of a single radar. 2D network is a special case of 3D one with $n_z = 0$. It tests discrete solutions in the trajectories' plane determined by points A, B and $(0, 0, 0)$ (radar position). We associate nodes of a 3D graph with integer vectors (i, j, k) forming 3D integer grid. Consequently, the set of arcs lengths in an integer grid with structure as shown in Figure 1.11 is $\{1, \sqrt{2}, \sqrt{3}, \sqrt{5}, \sqrt{6}, 3\}$. In assigning real arc length, all these values are scaled by an appropriate coefficient depending on actual size of a network. In order to reduce computational time, we approximated arcs lengths by the set of integer numbers $\{1000, 1414, 1732, 2236, 2449, 3000\}$. In this case, the scaling coefficient is adjusted correspondingly. Finding a constrained shortest path in a network with the integer lengths of arcs is approximately 1.5 times faster than the same procedure with real lengths of arcs. However, due to the integer approximation of arcs lengths, the actual length of a constrained shortest paths in a network may be slightly greater than assigned length constraints, while the corresponding optimal risk value may be lesser than the one obtained by analytical solutions approach. In tables presenting results of network flow optimization, optimal risk values, obtained by network optimization and inconsistent with "true" ones in the discussed sense, are marked by † symbol.

We calculated constrained shortest paths depending on ellipsoid shape (parameter κ) and length constraint, l_*, and compared optimal risk values R_{2D} and R_{3D} (in 2D and 3D network optimization, respectively) with the "true" ones, obtained by analytical solutions approach. Values of l_{2D} and l_{3D} are lengths of constrained shortest paths in 2D and 3D cases, respectively. We were interested in the following parameters: number of nodes left after preprocessing, N_{prep}, cost upper bound in preprocessing, U, preprocessing time, T_{prep}, number of labels treated in the LSA, N_{labels}, and running time of the LSA, T_{LSA}, measured in seconds. All these parameters are helpful in evaluating perfor-

mance of the discrete optimization approach. We analyzed the impact of using smoothing condition on accuracy of discrete solutions and LSA running time in 3D case.

5.1. 2D Network Optimization in the Case of a Single Radar

To calculate 2D optimal trajectories for the cases considered in Examples 2 and 3, we used a 2D squared graph with the following parameters

$$\begin{aligned}
\textit{Size of the graph} &= 2.3 \times 2.3 \\
n_x \times n_y &= 46 \times 46 \\
\textit{Length of axis arcs} &= 2.3/46 = 0.05 \\
\textit{Number of nodes} &= (46+1)^2 = 2209 \\
\textit{Number of arcs} &= 33672 \\
\textit{Radar position} &= (0,0) \\
\textit{Point } A &= (-0.25, 0.25) \\
\textit{Point } B &= (1.75, 0.25)
\end{aligned}$$

Numerical results of 2D network optimization for different values of κ and l_* are presented in Tables 1.3 and 1.4. Figures 1.14 and 1.15 compare analytical and discrete optimization solutions in 2D space with parameters: a) $n = 4$, $\kappa = 1.0$, $l_* = 2.6, 3.2, 4.0$; and b) $n = 4$, $\kappa = 0.1, 2.0$, $l_* = 3.2$, respectively. The smooth curves are the optimal trajectories obtained by the analytical approach and the nonsmooth curves are those obtained by solving the CSPP. Analytical and corresponding discrete optimization trajectories are close to each other, which validates both approaches. Note for the case of $k = 10.0$, values of optimal risk for discrete trajectories in Tables 1.3, 1.4, 1.5, 1.6 and 1.7 are lesser than the risk value for the corresponding analytical solution. Due to integer approximation of arcs lengths, the total length of those paths are greater than their integer representations.

Table 1.3. Results of 2D network preprocessing: single radar.

κ	l_*	*True Risk*	U	N_{pr}	T_{pr}, *sec*
1.0	2.6	9.792116	91.336509	837	0.359
1.0	3.2	8.421726	91.336509	1260	0.313
1.0	4.0	7.966210	35.693792	1703	0.500
0.1	3.2	0.468371	15.942982	1281	0.484
0.5	3.2	3.980716	66.836609	1265	0.329
2.0	3.2	12.464087	12.858975	764	0.547
10.0	3.2	14.845494	14.84242†	42	0.359

Table 1.4. Results of 2D network optimization with LSA: single radar.

κ	l_*	*True Risk*	l_{2D}	R_{2D}	N_{labels}	T_{LSA}, *sec*
1.0	2.6	9.792116	2.592	9.983805	85461	0.609
1.0	3.2	8.421726	3.1995	8.504245	224524	2.047
1.0	4.0	7.966210	3.9933	8.004489	423056	4.750
0.1	3.2	0.468371	3.199	0.488162	329517	3.641
0.5	3.2	3.980716	3.1932	4.063807	292594	2.922
2.0	3.2	12.464087	3.1958	12.518963	60619	0.406
10.0	3.2	14.845494	3.1958	14.84242†	0	0

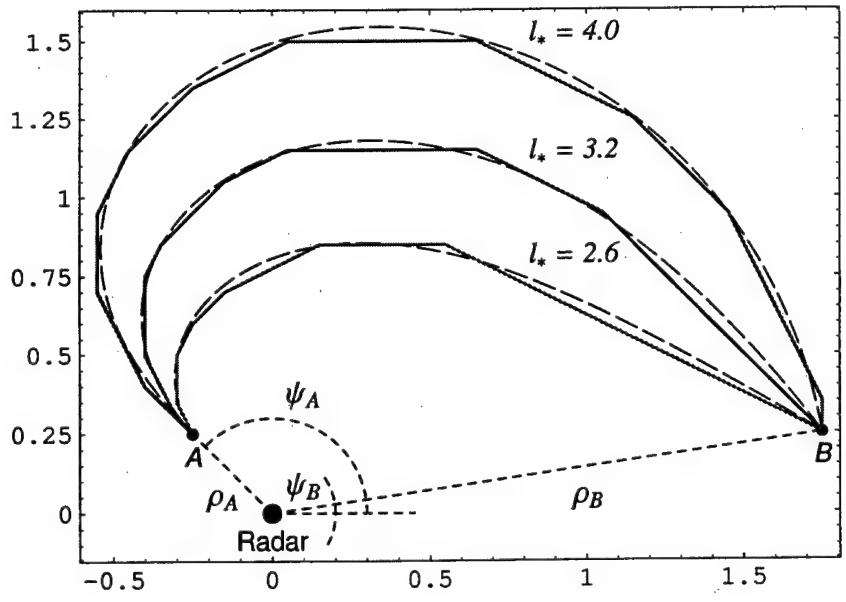

Figure 1.14. Comparison of analytical and discrete optimization trajectories for the case of sphere ($\kappa = 1.0$), $n = 4$ and different length constraints, l_*, in trajectories' plane.

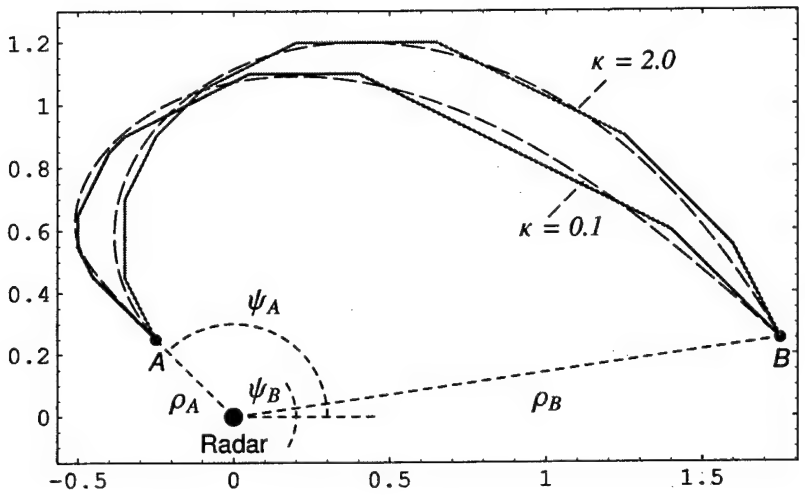

Figure 1.15. Comparison of analytical and discrete optimization trajectories for elongated ($\kappa = 0.1$) and compressed ($\kappa = 2.0$) ellipsoids for $n = 4$ and the same constraint on the length $l_* = 3.2$ in trajectories' plane.

5.2. 3D Network Optimization in the Case of a Single Radar

For 3D network optimization with the data from Examples 2 and 3, we used a parallelepiped graph with parameters

Size of the graph	=	$2.3 \times 1.0 \times 1.25$
$n_x \times n_y \times n_y$	=	$46 \times 20 \times 25$
Length of axis arcs	=	$2.3/46 = 1.0/20 = 1.25/25 = 0.05$
Number of nodes	=	$(46+1)(20+1)(25+1) = 25662$
Number of arcs	=	2213062
Radar position	=	$(0, 0, 0)$
Point A	=	$(-0.25, 0.15, 0.2)$
Point B	=	$(1.75, 0.15, 0.2)$

Numerical results of 3D network optimization with and without network structure smoothing for different values of κ and l_* are presented in Tables 1.5, 1.6 and 1.7. Figures 1.16, 1.17 and 1.18 compare the analytical and discrete optimization solutions in 3D space for the following sets of parameters: a) $n = 4$, $\kappa = 1.0$, $l_* = 2.6, 3.2, 4.0$; b) $n = 4$, $\kappa = 0.1$, $l_* = 3.2$; and c) $n = 4$, $\kappa = 2.0$, $l_* = 3.2$, respectively.

Table 1.5. Results of 3D network preprocessing: single radar.

κ	l_*	*True Risk*	U	N_{pr}	T_{pr}, sec
1.0	2.6	9.792116	91.336509	11518	9.516
1.0	3.2	8.421726	91.336509	22521	14.234
1.0	4.0	7.966210	8.902138	21175	18.031
0.1	3.2	0.468371	15.942982	22598	16.094
0.5	3.2	3.980716	66.836609	22553	14.875
2.0	3.2	12.464087	12.969952	15543	22.187
10.0	3.2	14.845494	14.840356†	2873	12.781

Table 1.6. Results of 3D network optimization with LSA: single radar.

κ	l_*	*True Risk*	l_{3D}	R_{3D}	N_{labels}	T_{LSA}, sec
1.0	2.6	9.792116	2.5998	10.251423	1936613	766.704
1.0	3.2	8.421726	3.19805	8.525182	6644608	6597.750
1.0	4.0	7.966210	3.9997	8.040955	8066613	7252.609
0.1	3.2	0.468371	3.1987	0.554746	9930869	11519.922
0.5	3.2	3.980716	3.1963	4.11088	8427875	9069.281
2.0	3.2	12.464087	3.19895	12.529767	2529266	1220.188
10.0	3.2	14.845494	3.1953	14.83876†	132135	8.031

Table 1.7. Results of 3D network optimization with LSA & smoothing condition: single radar.

κ	l_*	*True Risk*	l_{3D}	R_{3D}	N_{labels}	T_{LSA}, sec
1.0	2.6	9.792116	2.5998	10.251423	1080091	299.735
1.0	3.2	8.421726	3.19645	8.526340	4841651	4286.844
1.0	4.0	7.966210	3.9981	8.041533	6849799	5937.016
0.1	3.2	0.468371	3.19925	0.55662	5262314	5066.953
0.5	3.2	3.980716	3.1986	4.123165	5091681	4715.875
2.0	3.2	12.464087	3.1998	12.530477	2015458	846.672
10.0	3.2	14.845494	3.1953	14.83876†	76150	4.516

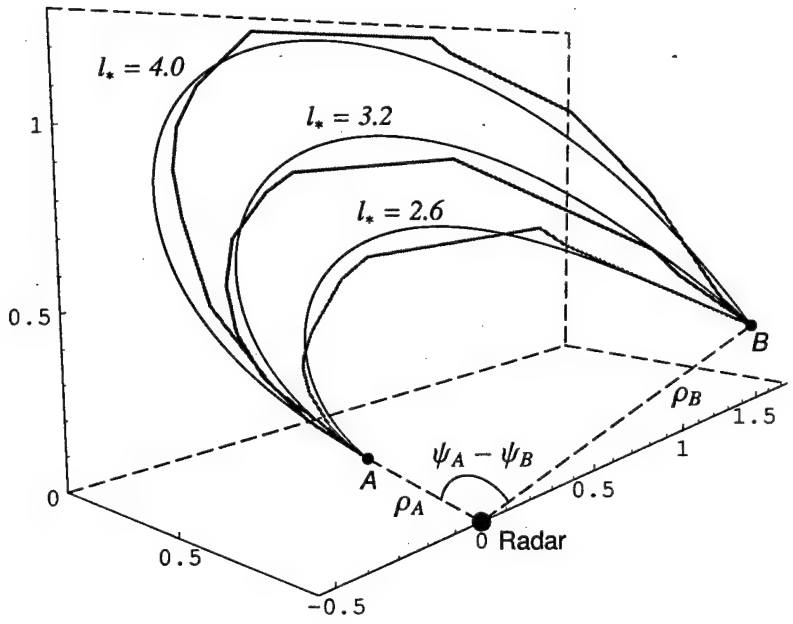

Figure 1.16. Comparison of analytical and discrete optimization trajectories for sphere ($\kappa = 1.0$), $n = 4$ with different length constraints, l_*, in 3D space.

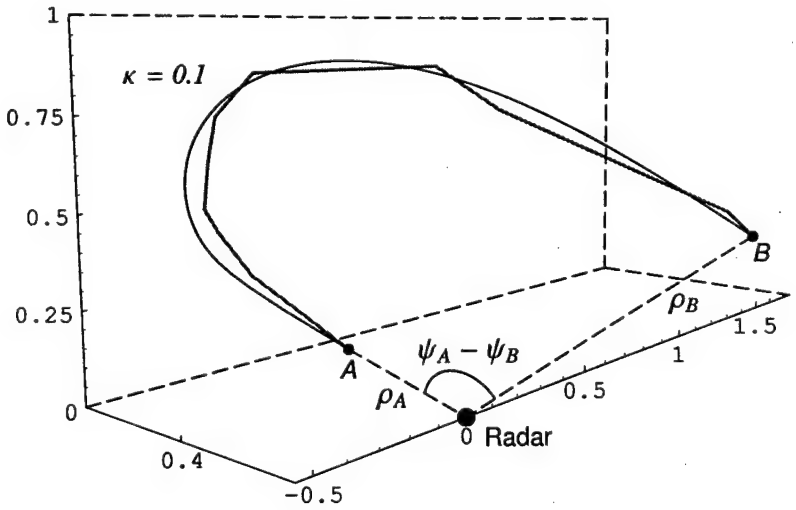

Figure 1.17. Comparison of analytical and discrete optimization trajectories for elongated ellipsoid $\kappa = 0.1$ and parameters $n = 4$, $l_* = 3.2$ in 3D space.

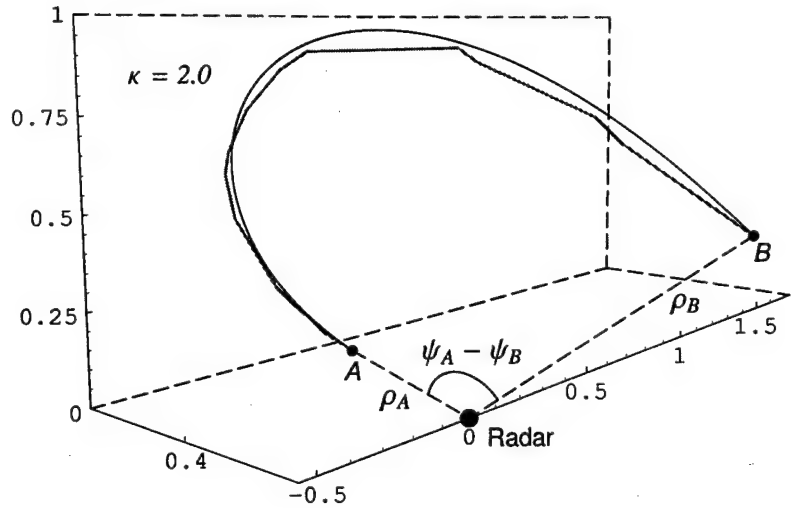

Figure 1.18. Comparison of analytical and discrete optimization trajectories for compressed ellipsoid $\kappa = 2.0$ and parameters $n = 4$, $l_* = 3.2$ in 3D space.

5.3. 3D Network Optimization in Cases with Two and Three Radars

This section analyzes impact of *variable* RCS in the case of several radars on: a) geomety of optimal trajectories; and b) performance of the discrete optimization approach. For optimal trajectory generation in the cases with two and three radars, we used the same 3D network of $2.3 \times 1.0 \times 1.25$ with the corresponding integer grid $n_x \times n_y \times n_y = 46 \times 20 \times 25$ and the following data for radars positions and staring and ending trajectory points

Radar 1	=	$(1, 0, 0)$	*Point A*	=	$(0, 0.5, 0)$
Radar 2	=	$(0.5, 1, 0)$	*Point B*	=	$(2, 0.5, 0)$

and

Radar 1	=	$(1, 0, 0)$	*Point A*	=	$(0, 0.75, 0)$
Radar 2	=	$(0.5, 1.25, 0)$	*Point B*	=	$(2, 0.75, 0)$
Radar 3	=	$(1.5, 1, 0)$			

Numerical results of 3D network preprocessing and optimization in the cases of two and three radars with and without network structure smoothing for different values of κ and the same constraint on the length, $l_* = 3.2$, are presented in Tables 1.8, 1.9, 1.10 and 1.11. Figures 1.19–1.24 illustrate discrete optimization trajectories in 3D space with two and three radars for the following parameters: $n = 4$, $l_* = 3.2$, $\kappa = 0.1, 1.0, 2.0$.

Table 1.8. Results of 3D network preprocessing: two radars

κ	l_*	U	N_{pr}	T_{pr}, sec
0.1	3.2	3.086443	20342	16.547
1.0	3.2	15.367339	19267	12.766
2.0	3.2	7.535904	15968	16.828

Table 1.9. Results of 3D network optimization: two radars (*smoothing is used)

κ	l_*	l_{3D}	R_{3D}	N_{labels}	T_{LSA}, sec
0.1	3.2	3.19885	0.921916	9993019	9837.906
1.0	3.2	3.1993	4.891124	6737166	5392.109
2.0	3.2	3.1993	4.320838	3617699	2010.750
*0.1	3.2	3.1993	0.922975	6428635	5556.891
*1.0	3.2	3.1993	4.891411	5095577	3708.437
*2.0	3.2	3.19785	4.330046	2818012	1335.859

Table 1.10. Results of 3D network preprocessing: three radars

κ	l_*	U	N_{pr}	T_{pr}, sec
0.1	3.2	19.374493	20766	14.718
1.0	3.2	28.740118	19660	16.641
2.0	3.2	22.262851	18377	15.625

Table 1.11. Results of 3D network optimization: three radars (*smoothing is used)

κ	l_*	l_{3D}	R_{3D}	N_{labels}	T_{LSA}, sec
0.1	3.2	3.19995	1.635309	7898013	6733.031
1.0	3.2	3.19905	9.073111	5623460	3998.969
2.0	3.2	3.19905	8.62298	3982280	2285.640
*0.1	3.2	3.19995	1.639800	5296123	4056.406
*1.0	3.2	3.197	9.094264	4413319	2916.281
*2.0	3.2	3.19905	8.62298	3155044	1634.031

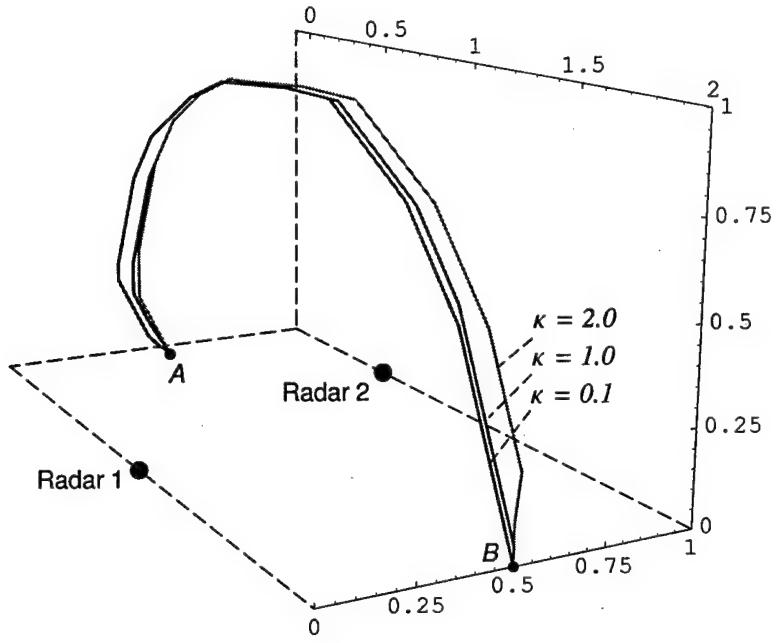

Figure 1.19. Optimal trajectories in the case of two radars for compressed ellipsoid ($\kappa = 2.0$), sphere ($\kappa = 1.0$) and elongated ellipsoid ($\kappa = 0.1$) with the same length constraint, $l_* = 3.2$.

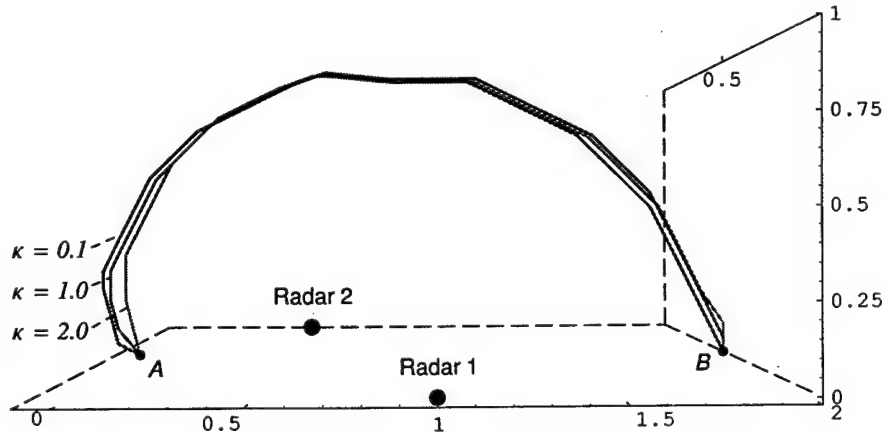

Figure 1.20. Front view: optimal trajectories in the case of two radars for compressed ellipsoid ($\kappa = 2.0$), sphere ($\kappa = 1.0$) and elongated ellipsoid ($\kappa = 0.1$) with the same length constraint, $l_* = 3.2$.

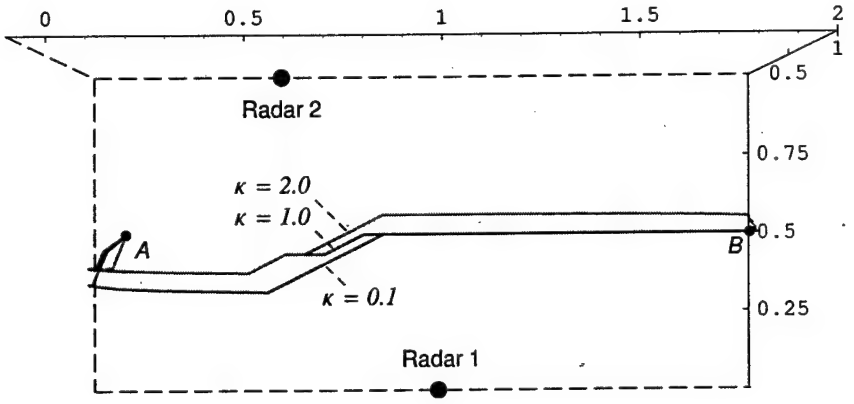

Figure 1.21. View from above: optimal trajectories in the case of two radars for compressed ellipsoid ($\kappa = 2.0$), sphere ($\kappa = 1.0$) and elongated ellipsoid ($\kappa = 0.1$) with the same length constraint, $l_* = 3.2$.

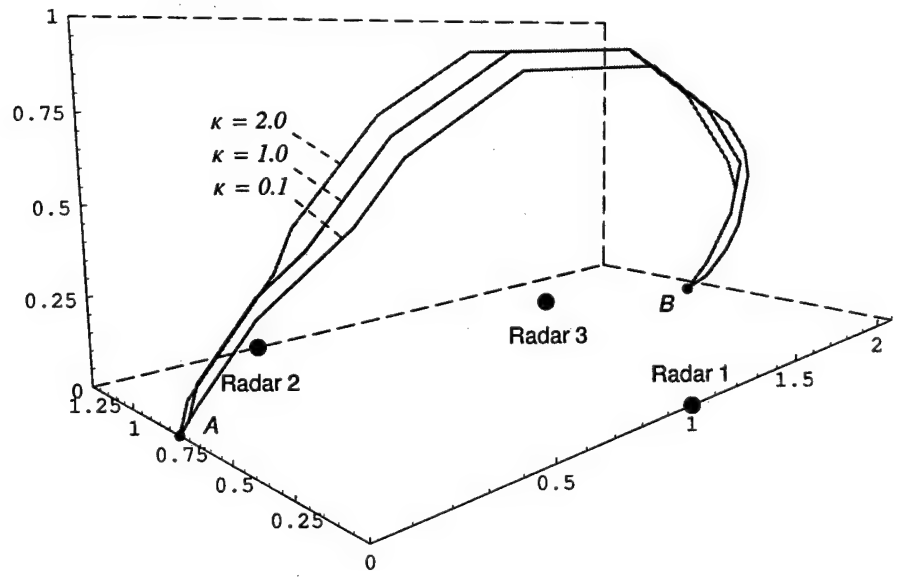

Figure 1.22. Optimal trajectories in the case of three radars for compressed ellipsoid ($\kappa = 2.0$), sphere ($\kappa = 1.0$) and elongated ellipsoid ($\kappa = 0.1$) with the same length constraint, $l_* = 3.2$.

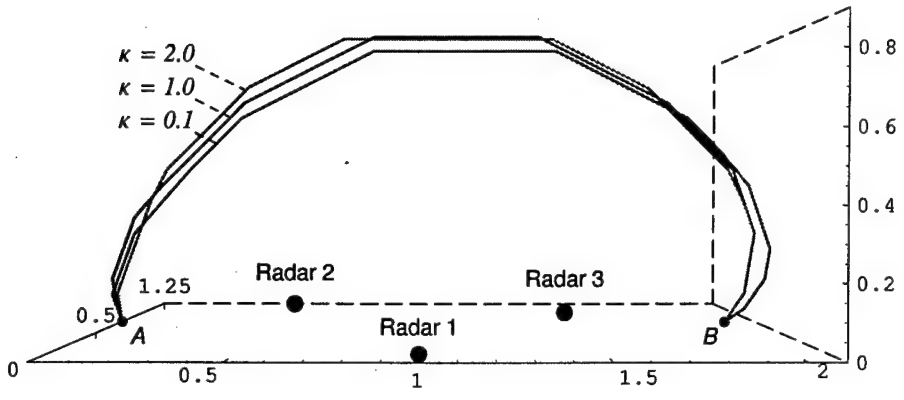

Figure 1.23. Front view: optimal trajectories in the case of three radars for compressed ellipsoid ($\kappa = 2.0$), sphere ($\kappa = 1.0$) and elongated ellipsoid ($\kappa = 0.1$) with the same length constraint, $l_* = 3.2$.

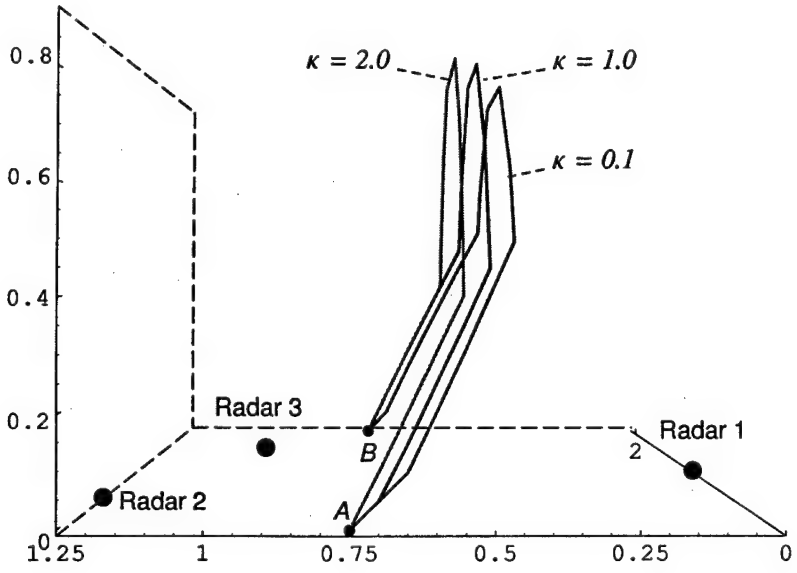

Figure 1.24. Side view: optimal trajectories in the case of three radars for compressed ellipsoid ($\kappa = 2.0$), sphere ($\kappa = 1.0$) and elongated ellipsoid ($\kappa = 0.1$) with the same length constraint, $l_* = 3.2$.

- *There is no strong correlation between LSA running time and number of radars; depending on ellipsoid shape it may decrease ($\kappa = 1.0$), increase ($\kappa = 2.0$) or variate ($\kappa = 1.0$).*

- *Running time of preprocessing procedure is always small (10–20 sec in 3D case), which in most testing examples is less than 2% of total computational time, and indicates no predictive power for LSA running time. However, number of nodes left after preprocessing is helpful in evaluating expected LSA running time. Also, testing examples suggest that LSA running time may linearly depend on number of treated labels. Although this number is known only after the algorithm stops, it can be used as a reference value for another run.*

- *In both examples with several radars, optimal trajectories for different values of parameter κ (ellipsoid shape), subject to the same constraint on trajectory length, are again close to each other (the same phenomenon was observed by comparing analytical trajectories in the case of a single radar).*

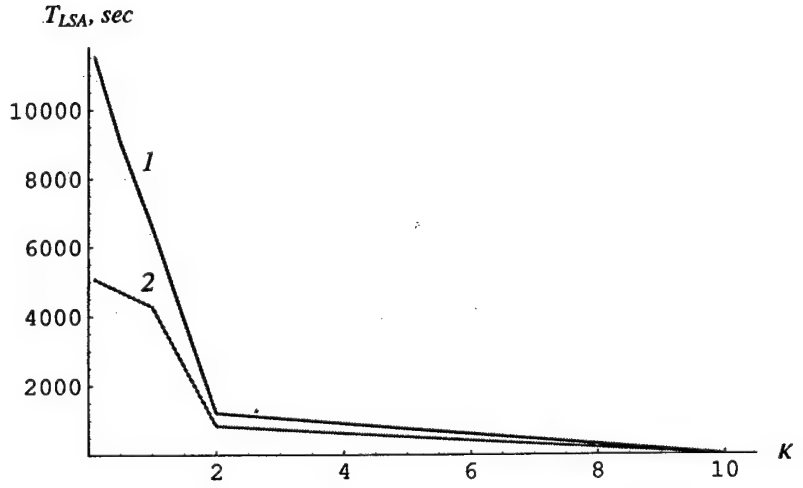

Figure 1.25. Dependence of LSA running time on the shape of ellipsoid, κ (3D network, single radar): curve "1" – no smoothing, curve "2" – smoothing is used.

Figure 1.25 shows dependence of LSA running time on the shape of ellipsoid, κ, in the case of 3D space and a single radar with and without network smoothing. The excessive running time for the LSA in the case of very elongated ellipsoids, ($\kappa << 1$, ellipsoid with $\kappa = 0.1$ is almost a needle), can be explained by lowest risk accumulations in directions radial to a radar, which

are ones producing greatest total lengths from point A to point B. This complicates comparison of labels in risk minimization, while balancing length constraint. This idea is supported by the fact that the running time in generating optimal trajectory for a compressed ellipsoid with $\kappa = 10.0$ is just several seconds. In this case, because of compressed geometry (for instance, a disk flying along its axis of symmetry), the risk of detection accumulates slower in directions transversal to a radar and those direction are the ones producing lowest total lengths from point A to point B. It is worth to mention that for small values of κ, network smoothing reduces LSA running time more efficiently.

Running time analysis. Figure 1.26 illustrates dependence of LSA running time, T_{LSA}, on number of labels treated, N_{labels}, in a 3D network for a single radar and various κ and l_* with and without smoothing. Variations of κ and l_* have no strong effect on LSA running time because of the preprocessing procedure. The running time is almost linearly depends on the number of labels treated, which, in turn, is a function of number of nodes left after preprocessing, N_{pr}, smoothing condition and cost upper bound, U, obtained in preprocessing. Figure 1.27 shows strong correlation between N_{labels} and N_{pr} plotted for all κ and l_* in the case of a single radar. While N_{labels} is uniformly reduced by smoothing condition, it may be quite different for the same value of N_{pr} because of different cost upper bounds obtained in preprocessing. Obviously, the closer U to optimal cost is, the lower number of labels will be treated. According to results presented in Tables 1.3, 1.5, 1.9 and 1.11, cost upper bounds are not close enough to "true" risk values. This fact suggests to develop preprocessing procedures obtaining more accurate cost upper bounds. Such preprocessing may be based on Lagrange relaxation [10, 11].

7. Conclusions

We developed a three-dimensional deterministic model for routing an aircraft with a variable radar cross-section (RCS) in a threat environment. The threat is associated with the risk of detection by radars, sensors or surface air missiles. To investigate dependence of the risk of detection on variable RCS, we model the aircraft by a symmetrical ellipsoid with the axis of symmetry orienting trajectory direction. The model considers the risk of detection to be the sum of risks from all installations in the area of interest, where the risk to be detected by a particular installation is proportional to the area of ellipsoid projection and reciprocal to the n^{th}-power of the distance between the aircraft and this particular installation. We developed analytical and discrete optimization approaches for solving the risk minimization problem subject to a constraint on trajectory length.

The analytical approach, based on calculus of variations, reduces the original problem to solving the vectorial nonlinear differential equation. We derived

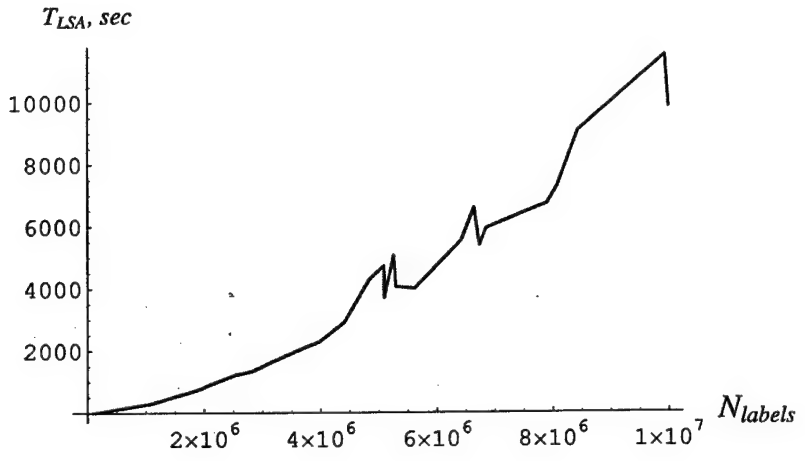

Figure 1.26. LSA running time versus number of labels treated: 3D network, single radar.

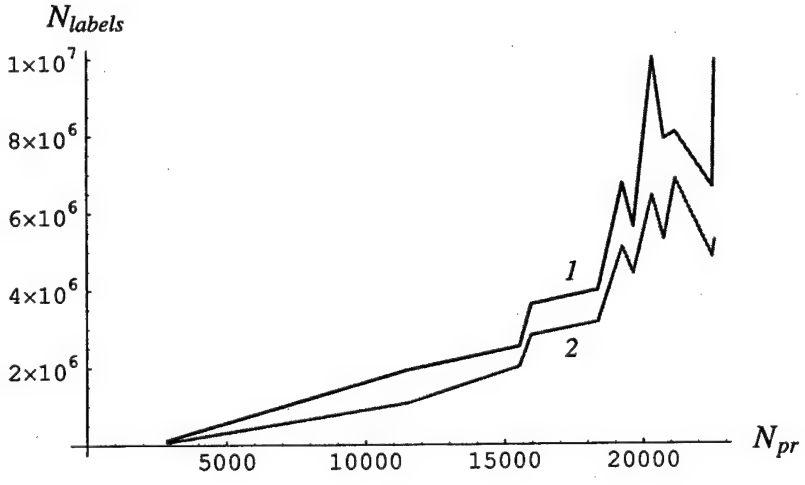

Figure 1.27. Number of labels treated versus number of nodes left after preprocessing (3D network, single radar): curves "1" and "2" correspond to LSA and LSA with smoothing, respectively.

this equation based on a general form of the risk functional in the case with an arbitrary number of passive or active installations. For the case of a single installation, arbitrary ellipsoid shape and any n, we obtained an analytical solution to the vectorial differential equation, which is expressed by a quadrature. According to numerical experiments based on the analytical solutions, we conclude that

- *Complexity of solving the vectorial differential equation analytically is primarily determined by the number of installations in the area of interest and is not affected by the type of an installation (radar or sensor).*
- *In the case of a single installation:*
  - *An analytical solution is expressed by a quadrature and corresponding optimal trajectory is a planar curve within the plane defined by starting and finishing trajectory's points and the radar's position.*
  - *The model with constant RCS ("spherical" aircraft, $\kappa = 1.0$) essentially simplifies obtaining an analytical solution and its further numerical analysis.*
- *The optimal risk is more sensitive to the variation of ellipsoid shape than to the variation of trajectory total length.*
- *Optimal trajectories for different κ (especially for $\kappa > 1$) are close to each other, which indicates that a variation of ellipsoid shape has no strong effect on the geometry of an optimal trajectory.*
- *Within proximity to an installation, an optimal trajectory is more sensitive to a radar-installation than to a sensor-installation and in area remote from the installation the effect is opposite.*

Obtaining an analytical solution to the vectorial differential equation in the case with an arbitrary number of installations is still an open issue. However, availability of an analytical solution in the case of a single installation significantly facilitates conceptual understanding the impact of variable RCS on the geometry of optimal trajectories and testing discrete optimization approaches.

To address optimal trajectory generation in 3D space in the case of variable RCS and arbitrary number of radars, we developed discrete optimization approach based on network flow optimization. Approximating the area of interest by a 3D network with a flexible structure and presenting a trajectory by a path in this network, NF optimization reduced optimal risk path generation with a constraint on trajectory length to the Constrained Shortest Path Problem (CSPP). We suggested to solve the CSPP by Label Setting Algorithm (LSA) with network smoothing, which is considered as an adjustment for the algorithm rather than a property of network structure. This condition, intended for

preserving trajectory smoothness and, as a result, eliminating inadmissible arcs in the network, can be used as a necessary constraint in trajectory generation. We tested NF optimization approach for 2D and 3D networks with and without smoothing condition, with various ellipsoid shapes, several constraints on trajectory length in the cases with one, two and three radars. Based on computational results of testing examples, we made the following conclusions

- *In the case of a single radar, all optimal trajectories obtained by the discrete approach for various κ and l_* are sufficiently close to the corresponding analytical trajectories.*
- *Network smoothing condition reduces LSA running time by 1.5–2.5 times, while preserving accuracy of optimal trajectories.*
- *LSA generates 2D discrete trajectories within a few seconds.*
- *LSA running time is extremely sensitive to the shape of ellipsoid; in 3D case, it varies from 5 to 5000 sec for* $0.1 \leq \kappa \leq 10.0$.
- *In testing examples with two and three radars, optimal trajectories with the same constraint on the length but different ellipsoid shapes are relatively close to each other, which suggests that in general, ellipsoid shape has no strong effect on the geometry of an optimal trajectory.*
- *Running time of the algorithm strongly depends on the value of trajectory length constraint.*

This chapter introduced 3D analytical model addressing optimal trajectory generation with variable RCS subject to a constraint on trajectory length. Developed analytical and discrete optimization approaches are just a first step in solving the proposed model rather than exhaustive answer to this matter. In 3D case, other NF algorithms for solving the CSPP as well as other approximation schemes for the original risk minimization problem may be addressed.

Appendix: Minimization of a Functional with Nonholonomic Constraint and Movable End Point

This section reduces necessary conditions for minimization of a functional with a nonholonomic constraint and a moveable end point to a vectorial nonlinear differential equation. This equation plays a central role in solving (5) in the case of a single radar. We consider the following general formulation

$$\min_{\mathbf{r}} \Phi(\mathbf{r}, \dot{\mathbf{r}}, l), \tag{1.A.1}$$

$$\Phi(\mathbf{r}, \dot{\mathbf{r}}, l) = \int_0^l L(\mathbf{r}(s), \dot{\mathbf{r}}(s))ds, \tag{1.A.2}$$

$$\mathbf{r}(0) = \mathbf{r}_1, \quad \mathbf{r}(l) = \mathbf{r}_2, \tag{1.A.3}$$

$$\varphi(\dot{\mathbf{r}}(s)) = 0, \tag{1.A.4}$$

$$l \le l_*, \tag{1.A.5}$$

where $\mathbf{r}(s) = (x(s), y(s), z(s))$, $\dot{\mathbf{r}}(s) = (\dot{x}(s), \dot{y}(s), \dot{z}(s))$.

A necessary condition for the existence of a functional extremum requires the total variation of the functional to be equal to zero. However, because of constraint (1.A.5), (1.A.1)–(1.A.5) is the problem with the movable end point, $\mathbf{r}(l)$, which means that variation of the total curve length, l, is not zero. Note variations δx, δy and δz are dependant by virtue of nonholonomic constraint (1.A.4). Lagrange multiplier method is used to separate differential expressions in the functional variation.

Let $\delta\mathbf{r} = (\delta x, \delta y, \delta z)$, $\delta\dot{\mathbf{r}} = (\delta\dot{x}, \delta\dot{y}, \dot{z})$, $\frac{\partial L}{\partial \mathbf{r}} = (\frac{\partial L}{\partial x}, \frac{\partial L}{\partial y}, \frac{\partial L}{\partial z})$ and $\frac{\partial L}{\partial \dot{\mathbf{r}}} = (\frac{\partial L}{\partial \dot{x}}, \frac{\partial L}{\partial \dot{y}}, \frac{\partial L}{\partial \dot{z}})$. Applying the Lagrange multiplier method to problem (1.A.1)–(1.A.2) with constraint (1.A.4), the functional (1.A.2) is rewritten as $\Phi(L, \phi, \lambda, l) = \int_0^l (L(\mathbf{r}, \dot{\mathbf{r}}) + \lambda(s)\,\phi(\dot{\mathbf{r}}))\,ds$. By definition the variation of this functional is

$$\begin{aligned}
\delta\Phi &= \int_0^l (\delta L(\mathbf{r}, \dot{\mathbf{r}}) + \lambda\,\delta\phi(\dot{\mathbf{r}}) + \phi\,\delta\lambda)\,ds + (L + \lambda\phi)|_{s=l}\,\delta l \\
&= \int_0^l \frac{\partial L}{\partial \mathbf{r}} \cdot \delta\mathbf{r} + \frac{\partial L}{\partial \dot{\mathbf{r}}} \cdot \delta\dot{\mathbf{r}} + \lambda \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \cdot \delta\dot{\mathbf{r}} + \phi\,\delta\lambda \;\; ds + (L + \lambda\phi)|_{s=l}\,\delta l \\
&= \int_0^l \frac{\partial L}{\partial \mathbf{r}} - \frac{d}{ds}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{d}{ds} \;\; \lambda \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad \cdot \delta\mathbf{r} + \phi\,\delta\lambda \;\; ds + \quad \frac{\partial L}{\partial \dot{\mathbf{r}}} + \lambda \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad \cdot \delta\mathbf{r} \;\; _{s=l} \\
&\quad + (L + \lambda\phi)|_{s=l}\,\delta l.
\end{aligned}$$

Note $\delta\mathbf{r}|_{s=l} \neq 0$, since l is varied and $s = l$ is not anymore a boundary point. Based on boundary conditions (1.A.3), the variation $\delta\mathbf{r}$ at the starting and finishing points $s = 0$ and $s = l + \delta l$, respectively, should be zero, i.e. $\delta\mathbf{r}(0) = 0$ and $\delta\mathbf{r}(l + \delta l) = 0$. The last condition is used to calculate the variation $\delta\mathbf{r}$ at $s = l$. Namely, from $\delta\mathbf{r}(l + \delta l) \equiv \delta\mathbf{r}(l) + \dot{\mathbf{r}}\delta l = 0$ we obtain $\delta\mathbf{r}(l) = -\dot{\mathbf{r}}\,\delta l$. Using the last equality, the variation $\delta\Phi$ is rearranged in the form

$$\begin{aligned}
\delta\Phi &= \int_0^l \frac{\partial L}{\partial \mathbf{r}} - \frac{d}{ds}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{d}{ds} \;\; \lambda \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad \cdot \delta\mathbf{r} + \phi\,\delta\lambda \;\; ds \\
&\quad + \;\; L + \lambda\phi - \;\; \frac{\partial L}{\partial \dot{\mathbf{r}}} + \lambda \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad \cdot \mathbf{r} \;\; _{s=l}\,\delta l.
\end{aligned}$$

Since after relaxing constraint (1.A.4), all three variations $(\delta x, \delta y, \delta z)$ became independent, the necessary conditions for an extremum, i.e. $\delta\Phi = 0$, are reduced to the constraint (1.A.4) and the following equations

$$\frac{\partial L}{\partial \mathbf{r}} - \frac{d}{ds}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{d}{ds} \;\; \lambda \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad = 0, \tag{1.A.6}$$

and

$$L - \;\; \frac{\partial L}{\partial \dot{\mathbf{r}}} + \lambda \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad \cdot \mathbf{r} \;\; _{s=l} = 0. \tag{1.A.7}$$

Vectorial equation (1.A.6) has the first integral. Indeed, the scalar product of (1.A.6) with $\dot{\mathbf{r}}$ gives

$$\frac{\partial L}{\partial \mathbf{r}} - \frac{d}{ds}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{d}{ds} \;\; \lambda \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad \cdot \dot{\mathbf{r}} = 0,$$

The left-hand side of this equality is a total differential, which after integration becomes

$$L - \dot{\mathbf{r}} \cdot \frac{\partial L}{\partial \dot{\mathbf{r}}} - \lambda \;\; \dot{\mathbf{r}} \cdot \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad = const. \tag{1.A.8}$$

Lagrange multiplier, $\lambda(s)$, is derived from (1.A.8)

$$\lambda(s) = \quad L - \dot{\mathbf{r}} \cdot \frac{\partial L}{\partial \dot{\mathbf{r}}} + c_L \qquad \dot{\mathbf{r}} \cdot \frac{\partial \phi}{\partial \dot{\mathbf{r}}} \quad , \tag{1.A.9}$$

where $c_L = -const$ is an unknown constant. Substitution of (1.A.9) into (1.A.6) leads to the vectorial differential equation for determining optimal $\mathbf{r}$

$$\frac{\partial L}{\partial \mathbf{r}} - \frac{d}{ds} \quad \frac{\partial L}{\partial \dot{\mathbf{r}}} + \frac{\frac{\partial \phi}{\partial \mathbf{r}}}{\dot{\mathbf{r}} \cdot \frac{\partial \phi}{\partial \dot{\mathbf{r}}}} \quad L - \dot{\mathbf{r}} \cdot \frac{\partial L}{\partial \dot{\mathbf{r}}} + c_L \quad = 0, \tag{1.A.10}$$

which along with the constraint (1.A.4) and boundary conditions (1.A.3) are necessary conditions for an extremum. Note that equations (1.A.10) and (1.A.4) are dependent in the sense that the scalar product of (1.A.10) with $\dot{\mathbf{r}}$ is reduced to $\ddot{\mathbf{r}} \cdot \frac{\partial \phi}{\partial \dot{\mathbf{r}}} = 0$, which is the differential of (1.A.4).

In the case when constraint $l \leq l_*$ is active, i. e. $l = l_*$, equation (1.A.7) is excluded from determining an optimal solution, since in this case curve total length is fixed and, therefore, the variation δl should equal zero by definition. If constraint $l \leq l_*$ is inactive, then from (1.A.7) and (1.A.8) we have $c_L = 0$.

However, (1.A.10) with (1.A.4) and (1.A.3) are only the necessary conditions for an optimal solution to solve minimization problem (1.A.1)–(1.A.5), since (1.A.10), (1.A.4) and (1.A.3) find an *extremal* trajectory, which either *minimizes* or *maximizes* the functional.

In the case of active length constraint (variation δl is zero), sufficient condition for an extremal, $\mathbf{r}^*$, minimizing functional (1.A.2) is formulated

for all $\mathbf{r}$ *sufficiently close to* $\mathbf{r}^*$ *and*

(a) *all* $\dot{\mathbf{r}}$ *sufficiently close to* $\dot{\mathbf{r}}^*$ *(weak minimum);*

(b) *all* $\dot{\mathbf{r}}$ *(strong minimum);*

the following relation holds

$$\int_0^{l_*} (L(\mathbf{r}, \dot{\mathbf{r}}) + \lambda^* \phi(\dot{\mathbf{r}}))\, ds \geq \int_0^{l_*} (L(\mathbf{r}^*, \dot{\mathbf{r}}^*) + \lambda^* \phi(\dot{\mathbf{r}}^*))\, ds,$$

where λ^* *is given by (1.A.9) calculated at* $\mathbf{r}^*$. This condition is reduced to verification of wether the second variation of the functional $\Phi(L, \phi, \lambda^*, l)$ at $\mathbf{r}^*$ is greater than or equal to zero.

References

[1] D. Assaf and A. Sharlin-Bilitzky. Dynamic search for a moving target. *Journal for Applied Probability*, 31(2):438–457, 1994.

[2] J. E. Beasley and N. Christofides. An algorithm for the resource constrained shortest path problem. *Networks*, 19:379–394, 1989.

[3] S. J. Benkoski, M. G. Monticino, and J. R. Weisinger. A survey of the search theory literature. *Naval Research Logistics*, 38(4):468–494, 1991.

[4] R. H. Bishop, D. L. Mackison, R. D. Culp, and M. J. Evans. *Spaceflight Mechanics 1999*, volume 102 of *Advances in the Astronautical Sciences*. NUnivelt, Inc., 1999.

[5] Y. K. Chan and M. Foddy. Real time optimal flight path generation by storage of massive data bases. In *Proceedings of the IEEE NEACON 1985*, pages 516–521, New York, 1985. Institute of Electrical and Electronics Engineers.

[6] M. Desrochers and F. Soumis. A generalized permanent labeling algorithm for the shortest path problem with time windows. *INFOR*, 26:191–212, 1988.

[7] I. Dumitrescu and N. Boland. Algorithms for the weight constrained shortest path problem. *ITOR*, 8:15–29, 2001.

[8] I. Dumitrescu and N. Boland. Improving preprocessing, labelling and scaling algorithms for the weight constrained shortest path problem. preprint, 2001.

[9] J. N. Eagle and J. R. Yee. An optimal branch-and-bound procedure for the constrained path, moving target search problem. *Operations research*, 38(1):110–114, 1990.

[10] G. Y. Handler and I. Zang. A dual algorithm for the constrained shortest path problem. *Networks*, 10:293–309, 1980.

[11] R. Hassin. Approximated schemes for the restricted shortest path problem. *Mathematics of Operations Research*, 17:36–42, 1992.

[12] B. O. Koopman. *Search and Screening: general principles with historical applications*. Pergamon Press, Elmsford, New York, 1980.

[13] M. Mangel. *Search Theory*. Springer-Verlag, Berlin, 1984.

[14] L. C. Polymenakos, D. P. Bertsekas, and J. N. Tsitsiklis. Implementation of effcient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, 43(2):278–283, 1998.

[15] A. Richards and J. P. How. Aircraft trajectory planning with collision avoidance using mixed integer linear programming. pages 1936–1941, May 2002.

[16] L. D. Stone. *Theory of Optimal Search*. Academic Press, New York, San Francisco, London, 1975.

[17] L. C. Thomas and J. N. Eagle. Criteria and approximate methods for path-constrained moving-target search problems. *Naval Research Logistics*, 42:27–38, 1995.

[18] J. N. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, 40(9):1528–1538, 1995.

[19] J. L. Vian and J. R. More. Trajectory optimization with risk minimization for military aircraft. *AIAA, Journal of Guidance, Control and Dynamics*, 12(3):311–317, 1989.

[20] A. R. Washburn. Search for a moving target: The fab algorithm. *Operations Research*, 31(3):739–751, 1983.

[21] A. R. Washburn. Continuous autorouters, with an application to submarines. Research Report NPSOR-91-05, Naval Postgraduate School, Monterey, CA, 1990.

[22] M. Zabarankin, S. Uryasev, and P. Pardalos. Optimal risk path algorithms. In R. Murphey and P. Pardalos, editors, *Cooperative Control and Optimization*, volume 66 of *Applied Optimization*, pages 271–303. Kluwer Academic Publishers, Dordrecht, 2002.

Part 2: Robust Decision Making: Addressing Uncertainties In Distributions

1. Introduction

This study develops a general approach to managing risk in military applications involving stochasticity and uncertainties in distributions. Various military applications such as intelligence, surveillance, planning, scheduling etc., involve decision making in dynamic, distributed, and uncertain environments. In a large system, multiple sensors may provide incomplete, conflicting, or overlapping data. Moreover, some components or sensors may degrade or become completely unavailable (failures, weather conditions, battle damage). Uncertainties in combat environment induce different kinds of risks that components, sensors or armed units are exposed to, such as the risk to be damaged or destroyed, risk of mission incompleteness (e.g., missing a target) or failure, risk of false target attack etc. Therefore, planning and operating in stochastic and uncertain conditions of a modern combat require robust decision-making procedures. Such procedures must take into account the stochastic nature of risk-inducing factors, and generate decisions that are not only effective on average (in other words, have good "expected" performance), but also safe enough under a wide range of possible scenarios. In this regard, risk management in military applications is similar to practices in other fields such as finance, nuclear safety, etc., where decisions targeted only at achieving the maximal expected performance may lead to an excessive risk exposure. However, in contrast to other applications, distributions of the stochastic risk-inducing factors are often unknown or uncertain in military problems. Uncertainty in distributions of risk parameters may be caused by a lack of data, unreliability of data, or the specific nature of a risk factor (e.g., in different circumstances a risk factor may exhibit different stochastic behavior). Therefore, decision making in military applications must account for uncertainties in distributions of stochastic parameters and be robust with respect to these uncertainties.

In this project, we propose a general methodology for managing risk in military applications involving various risk factors as well as uncertainties in distributions. We build our approach to risk management applications on the CVaR methodology, which is a relatively new development (Rockafellar and Uryasev, 2000, Rockafellar and Uryasev, 2001). The approach is tested with several stochastic versions of the Weapon-Target Assignment problem.

The report is organized as follows. Section 2 develops various formulations of the stochastic Weapon-Target Assignment (WTA) problem with CVaR constraints. Results of numerical experiments for one-stage and two-stage stochastic WTA problems are presented in Section 3. The

Conclusions section summarizes the obtained results and outlines the directions of future research. The Appendix presents key theoretical results on risk management using Conditional Value-at-Risk (CVaR) risk measure, and describes the general approach to controlling risk when distributions of risk factors are uncertain.

2. Stochastic Weapon-Target Assignment Problem

The Weapon-Target Assignment (WTA) problem considers the optimal assignment of weapons to targets so as to minimize the surviving value of targets. The WTA problem is used in planning environment that features a whole spectrum of uncertainties, such as the number and types of targets in the battle space, their positions, and the probability of a weapon to destroy a target (e.g. probability of kill). To generate robust decisions, one must account for these uncertainties and the corresponding risks. In this section we present two formulations of the stochastic Weapon-Target Assignment problem that address the uncertainties in a weapon's probability of kill and in the number of targets.

2.1. Deterministic WTA Problem

The generic formulation of the Weapon Target Assignment problem is as follows. Given the set of targets and set of available weapons, one must find the optimal assignment of weapons to targets, such that, for example, the damage to the targets is maximized, or the cost of the operation is minimized. The WTA formulation that maximizes the damage to the targets (see, for example, Manne, 1958, denBroeger et al., 1959, Murphey, 1999) leads to a non-linear programming problem with linear constraints (NLP), and is the subject of future research. In this study we adopt another setup, where the total cost of the mission (including battle damage or loss) is minimized, while satisfying constraints on mission accomplishment (i.e., destruction of all targets with some prescribed probabilities). We assume that different weapons have different costs and efficiencies, and, in general, each may have a "multishot" capacity so that it may attack more than one target. In the deterministic setup of the problem we include also the constraint that prescribes how many targets a single weapon can attack.

The deterministic WTA problem is

$$\min_{x} \quad \sum_{k=1}^{K} \sum_{i=1}^{I} c_{ik}\, x_{ik} \tag{1a}$$

subject to

$$\sum_{k=1}^{K} x_{ik} \leq m_i, \quad i = 1, ..., I, \tag{1b}$$

$$x_{ik} \leq m_i \, v_{ik}, \quad i = 1, ..., I, \quad k = 1, ..., K, \tag{1c}$$

$$\sum_{k=1}^{K} v_{ik} \leq t_i, \quad i = 1, ..., I, \tag{1d}$$

$$1 - \prod_{i=1}^{I} (1 - p_{ik})^{x_{ik}} \geq d_k, \quad k = 1, ..., K, \tag{1e}$$

$$x_{ik} \in \mathcal{Z}^+, \quad v_{ik} \in \{0, 1\},$$

where

x_{ik} is the number of shots to be fired by weapon i at target k;

v_{ik} $= 1$, if weapon i fires at target k, and $v_{ik} = 0$ otherwise;

c_{ik} is the cost (including the battle loss or damage) of firing one shot from weapon i at target k; c_k includes the relative value of target k with respect to all other targets;

m_i is the shots capacity for weapon i;

t_i is the maximal number of targets which can be attacked by weapon i;

p_{ik} is the probability of destroying target k by firing one shot from weapon i;

d_k is the minimal required probability for destroying target k;

$\mathcal{Z}$ is the set of integer numbers, and $\mathcal{Z}^+$ is the set of non-negative integers.

The objective function in this problem equals to the total cost of the mission. The first constraint, (1b), states that the munitions capacity of weapon i cannot be exceeded. The second and the third constraints (1c) and (1d) are responsible for not allowing weapon i to attack more than t_i targets, where $t_i \leq K$. The last constraint (1e) ensures that after all weapons are assigned, target k is destroyed with probability not less than d_k. Note that this non-linear constraint can be linearized:

$$\sum_{i=1}^{I} \ln(1 - p_{ik}) \, x_{ik} - \ln(1 - d_k) \leq 0. \tag{2}$$

In this way the deterministic WTA problem (1a) can be formulated as a linear integer programming (IP) problem.

2.2. One-Stage Stochastic WTA Problem with CVaR Constraints

In real-life situations many of the parameters in model (1a)–(1e) are not deterministic, but stochastic values. For example, the probabilities p_{ik} of destroying target k may depend upon battle situation, weather conditions, and so on, and consequently, may be treated as being uncertain. Similarly, the cost of firing c_{ik}, which includes battle loss/damage, may also be a stochastic parameter. The number of targets K may be uncertain as well.

First, we consider a one-stage Stochastic Weapon-Target Assignment (SWTA) problem, where the uncertainty is introduced into the model by assuming that probabilities p_{ik} are stochastic and dependent on some random parameter ξ:

$$p_{ik} = p_{ik}(\xi).$$

In accordance to the described methodology of managing uncertainties and risks in military applications, we model the stochastic behavior of probabilities p_{ik} using scenarios. Namely, probabilities $p_{ik}(\xi)$ take different values $p_{ik}(\xi_s) = p_{iks}$, $s = 1, ..., S$ under S different scenarios. Such a scenario set may be constructed, for example, by utilizing the historical observations of weapons' efficiency in different environments, or by using simulated data, experts' opinions etc.

To control risks we use Conditional Value-at-Risk (CVaR) approach. A general risk management approach with CVaR functions is described in Appendix. We replaced the last constraint in (1a) by a CVaR constraint, where the loss function takes a positive value if the probability of destroying target k is less than d_k:

$$L_k(x, \xi) = \sum_{i=1}^{I} \ln\Big(1 - p_{ik}(\xi)\Big)\, x_{ik} - \ln(1 - d_k), \tag{3}$$

and takes a negative value otherwise. The CVaR constraint with confidence level α bounds the (weighted) average of $(1 - \alpha) \cdot 100\%$ highest losses. In our case, allowing small positive values of loss function (3) for some scenarios implies that for these scenarios target k is destroyed with probability slightly less than d_k, which may still be acceptable from a practical point of view.

Except for the constraint on the target destruction probability, the one-stage Stochastic WTA problem is identical to its deterministic pre-

decessor:

$$\begin{aligned}
&\min_x \sum_{k=1}^{K}\sum_{i=1}^{I} c_{ik}x_{ik} \qquad (4)\\
&\text{subject to}\\
&\sum_{k=1}^{K} x_{ik} \le m_i, \quad i=1,...,I,\\
&x_{ik} \le m_i v_{ik}, \quad i=1,...,I, \quad k=1,...,K,\\
&\sum_{k=1}^{K} v_{ik} \le t_i, \quad i=1,...,I,\\
&\mathsf{CVaR}_\alpha\left[L_k(x,\xi)\right] \le C_k, \quad k=1,...,K.
\end{aligned}$$

Here α is the confidence level, C_k are some (small) constants, and all other variables and parameters are defined as before. As demonstrated in (11), for the adopted scenario model with probabilities p_{ik}, the CVaR constraint for the k-th target

$$\mathsf{CVaR}_\alpha\left[L_k(x,\xi)\right] \le C_k$$

is represented by a set of linear inequalities:

$$\begin{aligned}
&\sum_{i=1}^{I} \ln(1-p_{iks})\, x_{ik} - \ln(1-d_k) - \zeta_k \le w_{sk}, \quad s=1,...,S,\\
&\zeta_k + (1-\alpha_k)^{-1}S^{-1}\sum_{s=1}^{S} w_{sk} \le C_k, \qquad (5)\\
&\zeta_k \in \mathcal{R}, \quad w_{sk} \ge 0, \quad s=1,...,S, \quad k=1,...,K.
\end{aligned}$$

Thus, the one-stage Stochastic WTA problem can be formulated as a mixed-integer programming (MIP) problem:

$$\begin{aligned}
&\min_x \sum_{k=1}^{K}\sum_{i=1}^{I} c_{ik}\, x_{ik} \qquad (6)\\
&\text{subject to}\\
&\sum_{k=1}^{K} x_{ik} \le m_i, \quad i=1,...,I,\\
&x_{ik} \le m_i\, v_{ik}, \quad i=1,...,I, \quad k=1,...,K,\\
&\sum_{k=1}^{K} v_{ik} \le t_i, \quad i=1,...,I,
\end{aligned}$$

$$\sum_{i=1}^{I} \ln(1-p_{iks})\, x_{ik} - \ln(1-d_k) - \zeta_k \le w_{sk},$$
$$s = 1, ..., S, \quad k = 1, ..., K,$$
$$\zeta_k + (1-\alpha_k)^{-1} S^{-1} \sum_{s=1}^{S} w_{sk} \le C_k, \quad k = 1, ..., K,$$
$$x_{ik} \in \mathcal{Z}^+, \quad v_{ik} \in \{0,1\}, \quad \zeta_k \in \mathcal{R}, \quad w_{sk} \ge 0,$$
$$s = 1, ..., S, \quad i = 1, ..., I, \quad k = 1, ..., K.$$

Note that different values of probability p_{ik} represent the uncertainty in the distributions of stochastic parameters discussed in the previous section. Indeed, different values of probability p_{ik} imply different probability measures for the random variable associated with the event of destroying target k by firing one unit of munitions by weapon i. In effect, CVaR constraint (4) is a risk constraint that incorporates multiple probability measures.

2.3. Two-Stage Stochastic WTA Problem with CVaR constraints

In this section we consider a more complex, but also more realistic two-stage Stochastic WTA problem, where the uncertain parameter is the number of targets to be destroyed.

This problem is more realistic since it models the effect of target discovery as being dynamic; that is, not all targets are known at any single instance of time. To address this type of uncertainty, we need to modify our notation.

Consider I weapons are deployed in some bounded region of interest and interval of time T with the goal of finding targets and then, once found, attacking those targets. If we delay all assignments of weapon shots until to targets until the final time T, then we have a deterministic, "static" WTA problem as in (1a)–(1e). If, on the other hand, we assume that weapons have at least 2 opportunities to shoot during the interval T, then the WTA problem is dynamic. In the later case we have the opportunity to avoid expending all our shots at targets discovered early in T by explicitly modeling the number of undiscovered targets in the objective function.

Assume that K now represents the number of *categories* of targets (the targets may be categorized, for example, by their importance, vulnerability, etc).

We will assume the problem has 2 stages. That is, at any given point in time, we may always partition all targets into those thus far determined

and those that we conjecture to exist but have not yet found. Our conjecture may be based on evidence obtained by prior reconnaissance of the region of interest. At some arbitrary time $0 < \tau < T$ assume that there are n_k detected targets and η_k undetected targets in each category $k = 1, ..., K$. Thus we have two clearly identified stages in our problem: in the first stage one has to destroy the targets known at time τ, in the second stage one must destroy the targets that we conjecture will be found by time T. In other words, one needs to make an assignment of weapons that will allow for the destruction of the targets known at time τ while reserving enough munition capacity for destroying the targets we expect to find in $\tau < t < T$.

Setup of the two-stage stochastic WTA problem can be considered as a part of a moving horizon or quasi-multistage stochastic WTA algorithm, where the WTA problem with many time periods is solved by recursive application of a two-stage algorithm (Murphey, 1999).

To simplify the problem setup, we remove the constraint on the number of targets a single weapon can attack (the second and third constraints in problems (1a)), since this constraint makes the problem much too combinatorial. Also, we assume that the probabilities p_{ik} are known (*not* random), so that the only stochastic parameters in the two-stage SWTA problem are the numbers of undetected (second-stage) targets η_k, $k = 1, ..., K$.

We model the uncertainty in the number of targets at the second stage, by we introducing a scenario model, where under scenario $s \in \{1, ..., S\}$ there are $\eta_k(s) = \eta_{ks}$ undetected targets in category k.

The first- and second-stage decision variables are defined as follows:

x_{ik} is the number of munitions to be fired by weapon i at a single target in category k during the first stage;

$y_{ik}(s)$ is the number of munitions to be fired by weapon i at a single target in category k during the second stage scenario s.

Note that the same decision is made for all targets within a category, i.e., once weapon i fires, say, 2 missiles at a specific target in category k, it must fire 2 missiles at every other target in this category.

The recursive formulation of the two-stage stochastic WTA problem is

$$\min \left\{ \sum_{k=1}^{K} \sum_{i=1}^{I} n_k c_{ik} x_{ik} + E_\eta[Q(x, \eta)] \right\} \tag{7a}$$

subject to

$$\sum_{k=1}^{K} n_k\, x_{ik} \le m_i, \quad i = 1, ..., I, \tag{7b}$$

$$\sum_{i=1}^{I} \ln(1 - p_{ik})\, x_{ik} - \ln(1 - d_k) \le \varepsilon_{1k}, \quad k = 1, ..., K, \tag{7c}$$

$$\sum_{k=1}^{K} \varepsilon_{1k} \le C, \tag{7d}$$

$$x_{ik},\ \varepsilon_{1k} \ge 0, \quad i = 1, ..., I, \quad k = 1, ..., K,$$

where the recourse function $Q(x, \eta)$ is the solution of the problem

$$Q(x, \eta) = \min_{y} \left\{ \sum_{k=1}^{K} \sum_{i=1}^{I} \eta_k(s)\, c_{ik}\, y_{ik}(s) + M \sum_{i=1}^{I} \delta_i \right\} \tag{8a}$$

subject to

$$\sum_{k=1}^{K} (n_k x_{ik} + \eta_k(s)\, y_{ik}(s)) \le m_i + \delta_i, \quad \forall\, i, \tag{8b}$$

$$\sum_{i=1}^{I} \ln(1 - p_{ik})\, y_{ik}(s) - \ln(1 - d_k) - \zeta_k \le w_k(s),\ \forall\, k,\ s, \tag{8c}$$

$$\zeta_k + (1 - \alpha_k)^{-1} S^{-1} \sum_{s=1}^{S} w_k(s) \le \varepsilon_{2k}, \quad \forall\, k, \tag{8d}$$

$$\sum_{k=1}^{K} (\varepsilon_{1k} + \varepsilon_{2k}) \le C, \tag{8e}$$

$$y_{ik}(s),\ \delta_i \in \mathcal{Z}^+, \quad w_k(s),\ \varepsilon_{2k} \ge 0, \quad \zeta_k \in \mathcal{R}, \quad M \gg 1.$$

Let us discuss the recourse problem (7a)–(8e). As before, we minimize the total cost of the mission. The first constraint (7b) is the munitions capacity constraint. The second constraint, (7c), allows a first-stage target in category k to survive with (small) error ε_{1k}, and the third constraint (7d) bounds the sum of errors ε_{1k} by some (small) constant C.

In the recourse function (8a) the first constraint (8b) requires the weapon i to not exceed its munitions capacity while destroying the first- and second-stage targets. The possible infeasibility of the munitions capacity constraint can be relaxed using auxiliary variables δ_i that enter the objective function with cost coefficient $M \gg 1$. The second and third constraints (8c)-(8d) form a CVaR constraint that controls the failure

of destroying second-stage targets with the prescribed probabilities d_k. Similarly to the deterministic constraint in (7a), CVaR of failure to destroy a second-stage target in category k is bounded by (small) error variable ε_{2k}. The total sum of errors ε_{1k} and ε_{2k} at both stages is bounded by small constant C, which makes possible a tradeoff between the degree of mission accomplishment at the first and second stages.

The extensive form of the two-stage SWTA problem (7a)–(8a) is

$$\min \left\{ \sum_{k=1}^{K} \sum_{i=1}^{I} n_k \, c_{ik} \, x_{ik} + \frac{1}{S} \sum_{s=1}^{S} \sum_{k=1}^{K} \sum_{i=1}^{I} \eta_{ks} \, c_{ik} \, y_{ik}(s) + M \sum_{i=1}^{I} \delta_i \right\} \quad (9)$$

subject to

$$\sum_{k=1}^{K} (n_k \, x_{ik} + \eta_{ks} \, y_{ik}(s)) \leq m_i + \delta_i, \quad \forall i, \, s,$$

$$\sum_{i=1}^{I} \ln(1 - p_{ik}) \, x_{ik} - \ln(1 - d_k) \leq \varepsilon_{1k}, \quad \forall k,$$

$$\sum_{i=1}^{I} \ln(1 - p_{ik}) \, y_{ik}(s) - \ln(1 - d_k) \, - \zeta_k \leq w_{ks}, \quad \forall k, \, s,$$

$$\zeta_k + (1 - \alpha_k)^{-1} S^{-1} \sum_{s=1}^{S} w_{ks} \leq \varepsilon_{2k}, \quad \forall k,$$

$$\sum_{k=1}^{K} (\varepsilon_{1k} + \varepsilon_{2k}) \leq C,$$

$$x_{ik}, \, y_{ik}(s), \, \delta_i \in \mathcal{Z}^+, \quad w_{ks}, \, \varepsilon_{1k}, \, \varepsilon_{2k} \in \mathcal{R}^+ \quad \zeta_k \in \mathcal{R}, \quad M \gg 1.$$

The two-stage stochastic WTA problem is also a MIP problem.

3. Numerical results

In this section we present and discuss numerical results obtained for both one-stage and two-stage stochastic WTA problems. The algorithms for solving deterministic, one- and two-stage stochastic WTA problems were implemented in C++, and we used CPLEX 7.0 Callable Library to solve the corresponding IP and MIP problems. We used simulated data (sets of weapons and targets, the corresponding costs and probabilities etc.) for testing the implemented algorithms.

3.1. Single-stage deterministic and stochastic WTA problems

For the deterministic and one-stage stochastic WTA problems we used the following data:

- 5 targets ($K = 5$)
- 5 weapons, each with 4 shots ($I = 5,\ m_i = 4$)
- any weapon can attack any target ($t_i = 5$),
- probabilities p_{ik} and costs c_{ik} depend only on the weapon index i: $p_{ik} = p_i$, $c_{ik} = c_i$
- all targets have to be destroyed with at least probability 95% ($d_k = 0.95$)
- the confidence levels α_k in CVaR constraint are 0.90
- there are 20 scenarios ($S = 20$) for probabilities $p_{ik}(s)$ in the one-stage SWTA problem; all scenarios are equally probable.

According to the aforementioned, we used simulated data for probabilities p_{iks} and costs c_{ik}. It was assumed that probabilities $p_{iks} = p_{is}$ are uniformly distributed random variables, and the Fig. 3.1 displays the relation between the cost of missile of weapon i and its efficiency (i.e., probability to destroy a target):

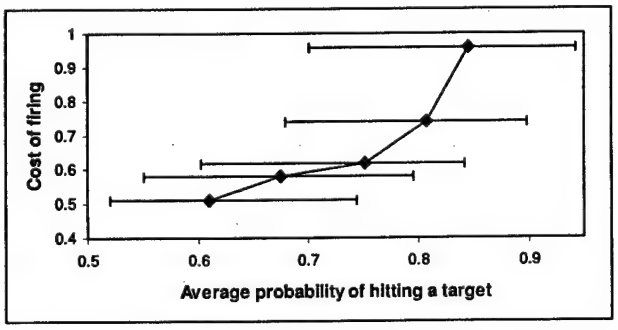

Figure 1.1. Dependence between the cost and efficiency for different types of weapons in one-stage SWTA problem (6) deterministic WTA problem (1a).

On this graph, diamonds represent the average probability of destroying a target by firing one shot from weapon i, and the horizontal segments

represent the support for random variable $p_{ik}(\xi) = p_i(\xi)$. The average probabilities

$$\overline{p}_{ik} = \frac{1}{S}\sum_{s=1}^{S} p_{iks}$$

were used for p_{ik} in the deterministic problem (1a).

The efficiency and cost of weapons 1 to 5 increase with the index of weapon, i.e., Weapon 1 is the least efficient and cheapest, whereas Weapon 5 is the most precise, but also most expensive one.

Tables 1.1 and 1.2 represent the optimal solutions (variables x_{ik}) of the deterministic and one-stage stochastic WTA problems.

Table 1.1. Optimal solution of the deterministic WTA problem (1a)

Target	T1	T2	T3	T4	T5	Total shots
Weapon 1	0	2	1	0	1	4
Weapon 2	0	1	2	0	0	3
Weapon 3	1	0	0	1	1	3
Weapon 4	1	0	0	1	1	3
Weapon 5	0	0	0	0	0	0

Table 1.2. Optimal solution of the one-stage stochastic WTA problem (4), (6)

Target	T1	T2	T3	T4	T5	Total shots
Weapon 1	0	1	1	0	1	3
Weapon 2	0	0	1	1	1	3
Weapon 3	2	0	0	1	0	3
Weapon 4	0	1	1	0	1	3
Weapon 5	1	1	0	1	0	3

One can observe the difference in the solutions produced by deterministic and stochastic WTA problems: the deterministic solution does *not* use the most expensive and most precise Weapon 5, whereas the stochastic solution of problem (6) with CVaR constraint uses this weapon. It means that the CVaR-constrained solution of problem (6) represents a more expensive but safer decision.

On a different dataset, we obtained a similar result: the optimal solution of the stochastic problem with CVaR constraints did not use the cheapest and the most unreliable weapon, whereas the deterministic solution used it.

We have also performed testing of the deterministic solution under different scenarios. The deterministic solution failed to destroy more than one target under 13 of 20 scenarios.

This example highlights the importance of using risk management procedures in military decision-making applications involving uncertainties.

3.2. Two-Stage Stochastic WTA Problem

For the two-stage stochastic WTA problems we used the following data:

- 3 categories of targets ($K = 3$)
- 4 weapons, each with 15 shots ($I = 4,\ m_i = 15$)
- probabilities p_{ik} and costs c_{ik} depend only on the weapon index i: $p_{ik} = p_i,\ c_{ik} = c_i$
- all targets have to be destroyed with probability 95% ($d_k = 0.95$)
- the confidence levels α_k in CVaR constraint are equal 0.90
- there are 15 scenarios ($S = 15$) for the number of undetected targets η_{ks} (for each k, the number of undetected targets η_{ks} is a random integer between 0 and 5); all scenarios are equally probable.

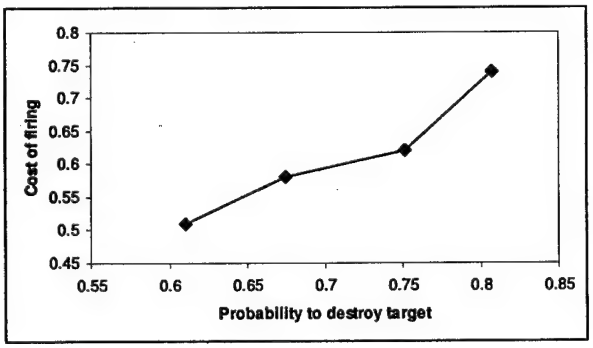

Figure 1.2. Dependence between the cost and efficiency for different types of weapons in two-stage SWTA problem (9).

For the probabilities p_{ik} in the two-stage problem, we used the first four average probabilities from the deterministic WTA problem, and the efficiency-cost dependence is shown in Fig. 3.2.

Tables 1.3 to 1.5 illustrate the optimal solution of the problem (9). Table 1.3 contains the first-stage decision variables x_{ik}, and Tables 1.4 and 1.5 display the second-stage variables $y_{ik}(s)$ for scenarios $s = 1$ and $s = 2$, just for illustrative purposes.

Similarly to the analysis of the one-stage stochastic WTA problem, we compared the scenario-based solution of problem (9) with the solution of the "deterministic two-stage" problem, where the number of second-stage targets in each category is taken as the average over 15 scenarios. The comparison shows that the solution based on the expected information leads to significant munitions shortages in 5 of 15 (i.e., 33%) scenarios, and consequently to failing the mission at the second stage. Recall from the analysis of the one-stage SWTA problem that the solution based on the expected information also exhibited poor robustness with respect to different scenarios. Indeed, solutions that use only the *expected* information, are supposed to perform well *on average*, or *in the long run*. However, in military applications there is *no long run*, and therefore such solutions may not be robust with respect to *many possible scenarios*.

Table 1.3. First-stage optimal solution of the two-stage stochastic WTA problem

Category	K1	K2	K3
# of detected targets	3	5	2
Weapon 1	0	0	0
Weapon 2	0	0	0
Weapon 3	1	1	1
Weapon 4	1	1	1

Table 1.4. First-stage optimal solution of the two-stage stochastic WTA problem (9) for the first scenario

Category	K1	K2	K3
# of undetected targets	1	4	2
Weapon 1	0	0	2
Weapon 2	0	0	1
Weapon 3	1	1	0
Weapon 4	1	1	0

Table 1.5. Second-stage optimal solution of the two-stage stochastic WTA problem (9) for the second scenario

Category	K1	K2	K3
# undetected of targets	3	5	3
Weapon 1	2	0	2
Weapon 2	1	0	1
Weapon 3	0	1	0
Weapon 4	0	1	0

Thus, solutions of both one-stage and two-stage SWTA problems confirm the general conjecture on the potential importance of exploiting stochastic models and risk management in military applications.

4. Conclusions

We have presented an approach to managing risk in stochastic environments, where distributions of stochastic parameters are uncertain. This approach is based on the methodology of risk management with Conditional Value-at-Risk risk measure developed by Rockafellar and Uryasev, 2000, 2001. Although the presented approach has been used to solve one-stage and two-stage stochastic Weapon-Target Assignment problems, it is quite general and can be applied to wide class of problems with risks and uncertainties in distributions. Among the directions of future research we emphasize consideration of a stochastic WTA problem in NLP formulation, where the damage to the targets is maximized while constraining the risk of false target attack.

References

Acerbi, C., and D. Tasche (2001) "On the Coherence of Expected Shortfall", *Working paper.*

Artzner, P., Delbaen, F., Eber, J.-M., and D. Heath (1999) "Coherent Measures of Risk", *Mathematical Finance*, 9, 203–228.

denBroeger, G. G., Ellison R. E., and L. Emerling (1959) " On optimum target assignments", *Operations Research*, 7, 322–326.

Golodnikov, A., Knopov, P., Pardalos, P., and S. Uryasev (2000) "Optimization in the Space of Distribution Functions and Applications in the Bayesian Analysis", *Research Report 2000-6, ISE Dept., Univ. of Florida.*

Manne, A. S. (1958) "A Target Assignment Problem", *Operations Research*, 6, 346–351.

Murphey, R. (1999) "An Approximate Algorithm For A Stochastic Weapon Target Assignment Problem", In: *Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems* (Pardalos, Ed.)

Rockafellar, R. T., and S. Uryasev (2000) "Optimization of Conditional Value-at-Risk", *Journal of Risk*, **2**, 21–41.

Rockafellar, R. T., and S. Uryasev (2001) "Conditional Value-at-Risk for General Loss Distributions", *Research Report 2001-5. ISE Dept., Univ. of Florida.*

5. Appendix. The General Approach to Risk Management

Presence of uncertainty in a decision-making model leads to the problem of estimation and managing/controlling of risk associated with the stochastic parameters in the model. Over the recent years, risk management has evolved into a sophisticated discipline combining both rigorous and elegant theoretical results and practical effectiveness (this especially applies to the risk management in finance industry). Generally speaking, risk management is a set of activities aimed at reducing or preventing *high losses* incurred from an incorrect decision. The losses (e.g., damages, failures) in a system are quantified by a *loss function* $L(x, \xi)$ that depends upon decision vector x and a stochastic vector ξ standing for uncertainties in the model. Assuming for now that a distribution of the parameter ξ is known, it is possible to determine the distribution of the loss function $L(x, \xi)$ (see Fig. 1.3). Then, the problem of preventing high losses is a problem of controlling and shaping the loss distribution and, more specifically, its right tail, where the high losses reside. To estimate and quantify the losses in the tail of the loss distribution, a *risk measure* has to be specified. In particular, a risk measure introduces the ordering relationships for risks, so that one is able to discriminate "less risky" decisions from the "more risky" ones[1]. The appropriate choice of a risk measure is, in most cases, dictated by the nature of uncertainties and risks in the problem at hand. In military applications, for example, one usually deals with the probabilities of events, such as the probability to hit a target, the probability to detect the enemy's aircraft,

[1]Artzner et al., 1999, have introduced a concept of "ideal", or *coherent*, risk measure. A *coherent* risk measure, which satisfies to a set of axioms developed in this paper, is expected to produce "proper" and "consistent" estimates of risk.

and so on. Therefore *percentile* risk measures that represent the risk in terms of percentiles of the loss distribution are particularly suitable for the risk management in military applications. Popular percentile risk measures include Value-at-Risk (VaR), Conditional Value-at-Risk, Maximum Loss, and Expected Shortfall. Figure 1.3 displays some of these measures; Value-at-Risk with confidence level α (α-VaR), which is the α-percentile of loss distribution, Maximum Loss ("1.0-percentile" of loss distribution), and α-CVaR, which may be thought of as the expectation of losses exceeding α-VaR.

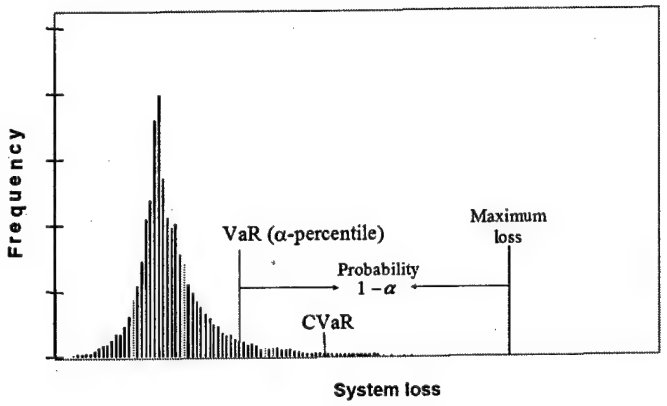

Figure 1.3. Loss function distribution and different risk measures.

We build our approach to risk management in military applications on the CVaR methodology, which is a relatively new development (Rockafellar and Uryasev, 2000, Rockafellar and Uryasev, 2001). This section presents the general framework of risk management using Conditional Value-at-Risk, and extends it to the case when the distributions of stochastic parameters are not certain.

5.1. Risk Management Using Conditional Value-at-Risk

Consider a loss function $L(x, \xi)$ depending on a decision vector x and a stochastic vector ξ, and its cumulative distribution function (c.d.f.) $\Psi(x, \zeta)$:

$$\Psi(x, \zeta) = \mathsf{P}[L(x, \xi) \leq \zeta].$$

Then the α-VaR (Value-at-Risk at confidence level α) function $\zeta_\alpha(x)$ corresponding to loss $L(x,\xi)$ is

$$\zeta_\alpha(x) = \min_{\zeta \in \mathcal{R}} \{\Psi(x,\zeta) \geq \alpha\}.$$

An "intuitive" definition of Conditional Value-at-Risk with confidence level α (α-CVaR) presents it as conditional expectation of losses exceeding the α-VaR level. This definition is correct, however, only for continuously distributed loss functions. For loss functions with general non-continuous distributions the α-CVaR function $\phi_\alpha(x)$ is defined as the expected value of random variable z_α (Rockafellar and Uryasev, 2001):

$$\phi_\alpha(x) = \mathsf{CVaR}_\alpha[L(x,\xi)] = \mathsf{E}[z_\alpha],$$

where c.d.f. $\Psi_{z_\alpha}(x,\zeta)$ of z_α has the form

$$\Psi_{z_\alpha}(x,\zeta) = \begin{cases} 0, & \zeta < \zeta_\alpha(x), \\ [\Psi(x,\zeta) - \alpha]/[1-\alpha], & \zeta \geq \zeta_\alpha(x). \end{cases}$$

In (Rockafellar and Uryasev, 2001), it was shown that α-CVaR can be expressed as a convex combination of α-VaR and conditional expectation of losses strictly exceeding α-VaR:

$$\phi_\alpha(x) = \lambda_\alpha(x)\,\zeta_\alpha(x) + [1 - \lambda_\alpha(x)]\,\phi_\alpha^+(x), \tag{10}$$

where

$$\phi_\alpha^+(x) = \mathsf{E}[L(x,\xi) \mid L(x,\xi) > \zeta_\alpha(x)], \tag{11}$$

which is also known as "upper CVaR" or Expected Shortfall, and

$$\lambda_\alpha(x) = [\Psi(x,\zeta_\alpha(x)) - \alpha]/[1-\alpha], \quad 0 \leq \lambda_\alpha(x) \leq 1.$$

Similar to (11), another percentile risk measure, called "lower CVaR", or CVaR$^-$, can be defined:

$$\phi_\alpha^-(x) = \mathsf{E}[L(x,\xi) \mid L(x,\xi) \geq \zeta_\alpha(x)].$$

Then, as it was shown in (Rockafellar and Uryasev, 2001), the introduced risk functions satisfy the following inequality:

$$\zeta_\alpha(x) \leq \phi_\alpha^-(x) \leq \phi_\alpha(x) \leq \phi_\alpha^+(x).$$

The conditional Value-at-Risk function $\phi_\alpha(x)$ has the following properties (Rockafellar and Uryasev, 2000, Rockafellar and Uryasev, 2001, Acerbi and Tasche, 2001):

- CVaR is continuous with respect to confidence level α (VaR, CVaR^+, CVaR^- may be discontinuous in α);
- CVaR is convex in α and x, provided that the loss function $L(x,\xi)$ is convex in x (VaR, CVaR^+, CVaR^- are generally non-convex in x);
- CVaR is *coherent* in the sense of Artzner et al., 1999;
- unlike VaR, CVaR has stable statistical estimates;
- in the case of a continuous loss distribution CVaR coincides with CVaR^+ and CVaR^-, and represents the conditional expectation of losses exceeding VaR.

From the viewpoint of managing and controlling of risk, the most important property of CVaR, which distinguishes it from all other percentile risk measures, is the convexity with respect to decision variables, which permits the use of convex programming for minimizing CVaR. If the loss function $L(x,\xi)$ can be approximated by a piecewise linear function, the procedure of controlling or optimization of CVaR is reduced to solving Linear Programming (LP) problem.

The techniques for optimizing CVaR when the loss distribution is discrete are of special importance for military applications, as will be demonstrated in Section 3.

Assume that there are S possible realizations (scenarios) $\xi_1, ..., \xi_S$ of vector ξ with probabilities π_s ($\sum_{s=1}^{S} \pi_s = 1$), then in the optimization problem with multiple CVaR constraints

$$\begin{aligned} &\max_{x \in X} \; g(x) \\ &\text{subject to} \\ &\mathsf{CVaR}_{\alpha_n}[L(x,\xi)] \leq C_n, \quad n = 1, ..., N, \end{aligned}$$

where $g(x)$ is some performance function and X is a convex set, each CVaR constraint may be replaced by a set of inequalities

$$\begin{aligned} &L(x,\xi_s) - \zeta_n \leq w_{ns}, \quad s = 1, ..., S, \\ &\zeta_n + (1-\alpha_n)^{-1} \sum_{s=1}^{S} \pi_s w_{ns} \leq C_n, \\ &\zeta_n \in \mathcal{R}, \quad w_{ns} \in \mathcal{R}^+, \quad s = 1, ..., S, \end{aligned} \tag{12}$$

where $\mathcal{R}$ is the set of real numbers, and $\mathcal{R}^+$ is the set of non-negative real numbers, and w_{ns} are auxiliary variables. If in the optimal solution

the n-th CVaR constraint is active, then the corresponding variable ζ_n is equal to α_n-VaR (i.e., α_n-th percentile of the loss distribution).

We also note that in the case when the behavior of stochastic parameter ξ can be represented by a scenario model $\{\xi_s,\ s = 1, ..., S\}$ with equally probable scenarios ($\pi_s = 1/S$), the concept of CVaR acquires especially simple and transparent interpretation. Namely, if (for a fixed x) the scenarios $\xi_1, ..., \xi_S$ are indexed such that $L(x, \xi_1) \leq \ldots \leq L(x, \xi_S)$, then α-CVaR equals the weighted average of losses for the $\lceil (1-\alpha)S \rceil$ worst scenarios:

$$\phi_\alpha(x) = \frac{1}{(1-\alpha)\,S}\left[(s_\alpha - \alpha S)\,L(x, \xi_{s_\alpha}) + \sum_{s=s_\alpha+1}^{S} L(x, \xi_s)\right],$$

where number s_α is such that

$$S - s_\alpha \leq (1-\alpha)S < S - s_\alpha + 1.$$

In the risk management methodology discussed above the distribution of stochastic parameter ξ is considered to be known. The next subsection extends the presented approach to the case, when the distribution of stochastic parameters in the model is not certain.

5.2. Risk Management Using CVaR in the Presence of Uncertainties in Distributions

The general approach to managing risks in an uncertain environment, where the distributions of stochastic parameters are not known for sure, can be described as follows. Suppose that we have some performance function $F(x, \xi)$, dependent on the decision vector $x \in X$ and some random vector $\xi \in \Xi$, whose distribution is not known for certain. We assume that the actual realization of vector ξ may come from different distributions $\Theta_1, ..., \Theta_N$. The vector ξ stands for the uncertainties in data that make it impossible to evaluate the efficiency $F(x, \xi)$ of the decision for sure. Thus, there always exists a possibility of making an incorrect decision, and, consequently, suffering loss, damage, or failing the mission. If the loss in the system is evaluated by function $L(x, \xi)$, then risk of high losses can be controlled using CVaR constrains. Let formulate the problem of maximizing the expected performance function $F(x, \xi)$ subject to some operational constraints $Ax \leq b$ and CVaR risk constraints. Due to the unknown distribution of vector ξ, we are unable to find the expectation $\mathsf{E}_\Theta[F(x, \xi)]$. Therefore, being on the conservative side, we want the decision x to be optimal with respect to each measure

Θ_n, and this leads to the following *maxi-min* problem:

$$\begin{aligned} &\max_{x \in X} \min_{\Theta_n,\, n=1,\ldots,N} \mathsf{E}_{\Theta_n}[F(x,\xi)] \\ &\text{subject to} \\ &Ax \le b, \\ &\mathsf{CVaR}_\alpha[L(x,\xi) \,|\, \Theta_n] \le C, \quad n = 1,\ldots,N, \end{aligned} \tag{13}$$

where multiple CVaR constraints with respect to different measures Θ_n control the risk for high losses $L(x,\xi)$ to exceed some threshold C. In formulation (13) we assume that the performance function F is concave in x, and the loss function L is convex in x. These assumptions are not restrictive; on the contrary, they indicate that given more than one decision with equal performance one favors safer decisions over the riskier ones.

Model (13) explains how to handle the risk of generating an incorrect decision in an uncertain environment. In military applications, different types of risks and losses may be explicitly involved, for example, along with loss function $L(x,\xi)$ one may consider a loss function $R(x,\xi)$ for the risk of false target attack. Control for this type of risk can also be included in the model by a similar set of CVaR constraints:

$$\begin{aligned} &\max_{x \in X} \min_{\Theta_n,\, n=1,\ldots,N} \mathsf{E}_{\Theta_N}[F(x,\xi)] \\ &\text{subject to} \\ &Ax \le b, \\ &\mathsf{CVaR}_{\alpha_1}[L(x,\xi) \,|\, \Theta_n] \le C_1, \quad n = 1,\ldots,N, \\ &\mathsf{CVaR}_{\alpha_2}[R(x,\xi) \,|\, \Theta_n] \le C_2, \quad n = 1,\ldots,N. \end{aligned}$$

In the next sections we test the presented approach to risk management in military applications on the Weapon-Target Assignment problem.

Part 3: Properties of No-Depot Min-Max 2-Traveling-Salesmen Problem

1. Introduction

Let $G(V, E)$ be an undirected graph with set of vertexes $V = \{1, 2, \ldots, n\}$ and set of edges E. Let $d(i, j)$ denote the length of edge (i, j). We assume that graph $G(V, E)$ is complete, *i.e.* any two vertexes are connected by an edge. A *path* on graph $G(V, E)$ is defined as an ordered subset of set V. The length of path $P = \{p_1, \ldots, p_r\}$ is defined by the following formula:

$$L(P) = \sum_{i=1}^{r-1} d(p_i, p_{i+1}) \tag{1}$$

A path that returns to the origin is a *cycle*. A cycle that visits each vertex of graph $G(V, E)$ once is a *tour* on graph $G(V, E)$. A tour $(t_1 \to t_2 \to \ldots \to t_r \to t_1)$ is defined (and denoted) by an ordered set $T = \{t_1, t_2, \ldots, t_r\}$. The length of tour T is computed by the following formula:

$$L(T) = \sum_{i=1}^{r-1} d(t_i, t_{i+1}) + d(t_r, t_1) \tag{2}$$

A cyclic rotation of vertexes in a tour does not change the length of the tour. A regular *Traveling Salesman Problem* (TSP) consists in finding the shortest tour on graph $G(V, E)$:

$$L_{TSP}(N) = \min_{T:T\subset V,|T|=n} L(T) \tag{3}$$

Multi-Traveling Salesmen Problem is an extension of TSP to the case of several salesmen. The objective of the Multi-Traveling Salesmen Problem with m salesmen (m-TSP) is to find a decomposition of graph $G(V, E)$ into m disjoint complete subgraphs minimizing an aggregated objective resulted from the particular solutions of TSPs on the subgraphs. *Min-Sum m-TSP with a depot vertex* is a prevalent formulation of m-TSP. According to this formulation, the objective is to determine m tours of the least total length, so that the depot vertex is present in every tour, and every other vertex of graph $G(V, E)$ is present in only one tour. This variation of m-TSP can be transformed into a regular TSP by introduction of artificial vertexes [1]. Unfortunately, the resultant TSP becomes very degenerate in most cases. The transformation of Min-Sum m-TSP with different depot vertexes appears to be difficult except for a special case of only two salesmen. The transformation for $m = 2$ is provided in [5].

A typical solution of Min-Sum m-TSP is highly irregular. The computational experience shows that the distribution of assignments is not uniform in the optimal solution: the difference in the length of two tours can be excessive. In many contexts this can be inappropriate: Giust [4] considers an example

of gas distribution by a small company that has four delivery cars. In another example, France *et al.* [2] describe a problem of scheduling n jobs to m identical machines in order to minimize the total time when processing times are job-sequence dependent. The authors consider the Min-Max m-TSP formulation with a depot vertex. In this formulation the objective is to determine m tours with the least length of the longest tour; every tour starts and ends at the depot vertex, and every vertex of graph $G(V, E)$ (except the depot vertex) is present in only one tour. The authors propose one heuristic and two exact search schemes for the Min-Max m-TSP with depot formulation.

We consider *Min-Sum m-TSP without a depot vertex* (No-Depot Min-Max m-TSP). In this case, the Submarine Routing Problem can be considered as a possible application of this formulation. In this problem graph $G(V, E)$ represents a region that needs to be monitored. Each vertex of the graph corresponds to a specific geographical location in the region. There is a fleet of submarines available for monitoring. Due to the high price of a submarine, the fleet is very limited, and usually contains 2-3 submarines. A patrol cycle of a submarine is the time needed to visit all the assigned locations and to return to the origin. The objective of the Submarines Routing Problem is to assign a specific route (tour) to each submarine, so that each location is visited once by only one submarine, and the longest patrol cycle is minimized. The problem can be formulated as follows:

$$L_{MIN-MAX} = \min_{M_1,\ldots,M_m} \max_{i\in\{1,\ldots,m\}} \{L_{TSP}(M_i)\} \tag{4}$$

$$V = M_1 \cup \ldots \cup M_m, \tag{5}$$

$$M_i \cap M_j = \emptyset \; \forall i \neq j \tag{6}$$

In this formulation of Min-Max m-TSP a depot vertex is not specified. The submarines are assigned to their patrol routes for many cycles for a period of time ranging from 1 to 5 years. Provision and crew changes are provided in several locations on the patrol routes by special ships or aircrafts. Usually, these locations are chosen after the submarines' routes are specified.

The purpose of this study is to study No-Depot Min-Max 2-TSP. We introduce a notion of characteristic function for this class of problems. Using constant graphs we study a connection between No-Depot Min-Max 2-TSP and a subclass of self-dual monotonic Boolean functions.

2. Characteristic Function for No-Depot Min-Max 2-TSP

We consider Min-Max No-Depot 2-TSP on complete undirected graph $G(V, E)$ with non-negative lengthes of edges. We consider a two-stage solution of the problem. At the first stage, the partition of set $V = \{1, \ldots, n\}$

into two subsets M_1 and M_2 is created. The first set M_1 is assigned to the first salesman, and the second set M_2 is assigned to the second salesman. At the second stage each salesman solves a regular TSP on the subgraph assigned to him. We associate each graph partitioning $\{M_1, M_2\}$ with a Boolean vector $\vec{\alpha} = \{\alpha_1, \ldots, \alpha_n\} \in \{0,1\}^n$ in the following way:

$$\alpha_i = \begin{cases} 0 & \text{if } i \in M_1 \\ 1 & \text{if } i \in M_2 \end{cases} \tag{7}$$

Further we will identify each decomposition $\{M_1, M_2\}$ of graph $G(V, E)$ by Boolean vector $\vec{\alpha} = \{\alpha_1, \ldots, \alpha_n\}$ according to rule (7).

Definition 2.1. *Characteristic function $f(x_1, \ldots, x_n)$ for No-Depot Min-Max 2-TSP on graph $G(V, E)$ (or simply a characteristic function for graph $G(V, E)$) is defined by the following rule:*

$$f(\vec{\alpha}) = \begin{cases} 1, & L_{TSP}(M_1) \geq L_{TSP}(M_2) \\ 0, & L_{TSP}(M_2) < L_{TSP}(M_2), \end{cases} \tag{8}$$

where $\vec{\alpha}$ is defined in (7).

Boolean vector $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is called *one* if $f(\vec{\alpha}) = 1$. One $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is called a *lower one* if there is no other one $\vec{\beta} = (\beta_1, \ldots, \beta_n)$ so that $\beta_i \leq \alpha_i \; \forall i = 1, \ldots, n$. Otherwise one $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is called *a generated one*. Similarly, Boolean vector $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is called *zero* if $f(\vec{\alpha}) = 0$. Zero $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is called an *upper zero* if there is no other zero $\vec{\beta} = (\beta_1, \ldots, \beta_n)$ so that $\beta_i \geq \alpha_i$ for $i = 1, \ldots, n$. Otherwise zero $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is called *a generated zero*.

A graph is called *metric* if all the vertexes in the graph correspond to the points in metric space, and the lengths of the edges equal the metric distances between the corresponding points. By the definition, any metric graph is a symmetric graph, and for any three graph vertexes i_1, i_2, and i_3 the triangle inequality $d(i_1, i_3) \leq d(i_1, i_2) + d(i_1, i_2)$ is satisfied.

Definition 2.2. *Graph $G(V, E)$ is splittable if for any three vertexes the triangle inequality is satisfied, and for its characteristic function there is no decomposition $\{M_1, M_2\}$ of the graph , for which $L_{TSP}(M_1) = L_{TSP}(M_2)$.*

(Note, in this definition it is not necessary that the considered graph is metric)

Statement 2.1. *The characteristic function of a metric (splittable) graph is monotonic.*

Indeed, since the triangle inequality holds for any three vertexes of a metric (splittable) graph, adding a new vertex to M_1 does not decrease $L_{TSP}(M_1)$, and removing a vertex from M_2 does not increase $L_{TSP}(M_2)$.

A Boolean function $f(\vec{\alpha})$ is self-dual if for any Boolean vector $\vec{\gamma^+} = (\gamma_1, .., \gamma_n)$ and vector $\vec{\gamma^-} = (1 - \gamma_1, .., 1 - \gamma_n)$

$$f(\vec{\gamma^+}) + f(\vec{\gamma^-}) = 1$$

The next property follows directly from definitions 2.1 and 2.2:

Statement 2.2. *The characteristic function of a splittable graph is self-dual.*

Consider a non-splittable metric graph. For this graph, the equality

$$L_{TSP}(M_1) = L_{TSP}(M_2)$$

holds for at least one decomposition $\{M_1, M_2\}$ of graph $G(V, E)$. The same equation holds for opposite decomposition $\{M_2, M_1\}$:

$$L_{TSP}(M_2) = L_{TSP}(M_1).$$

Therefore, the values of the characteristic function are equal on the vectors corresponding to decompositions $\{M_1, M_2\}$ and $\{M_2, M_1\}$. Hence, the characteristic function of the not-splittable metric graph is not self-dual. This situation can be easily avoided by small variation of distances in the graph.

Theorem 2.1. *If No-Depot Min-Max 2-TSP is considered for a metric (splittable) graph $G(V, E)$, at least one optimal solution of this problem belongs to the set of lower ones of the corresponding characteristic function. There is another optimal solution that belongs to the set of upper zeros.*

Proof. Suppose, that the characteristic function has neither upper zero, nor lower one, that corresponds to an optimal solution of No-Depot Min-Max 2-TSP (4)-(6). Therefore, every optimal solution of the problem corresponds either to generated zero, or to generated one. Because of the symmetry of the problem, the solution formed by the opposite decomposition is also optimal. If the original solution corresponds to a zero (one) of the characteristic function, the opposite solution corresponds to a one (zero) of this function. Consider optimal solution $\vec{\alpha^0} = (\alpha_1^0, \ldots, \alpha_n^0)$ that corresponds to a zero of characteristic function $f(\vec{\alpha})$:

$$\alpha_i^0 = \begin{cases} 0 & \text{if } i \in M_1 \\ 1 & \text{if } i \in M_2 \end{cases},$$

$$M_1 \cup M_2 = V, M_1 \cap M_2 = \emptyset.$$

According to the assumption, $\vec{\alpha^0}$ is a generated zero. Therefore, there exists an upper zero $\vec{\alpha^*}$ that exceeds $\vec{\alpha^0}$ in several components. So, there is a non empty set of vertexes H, $H \subset M_2$, so that:

$$\alpha_i^* = \begin{cases} 0 & \text{if } i \in M_1 \cup H \\ 1 & \text{if } i \in M_2 \setminus H \end{cases},$$

According to Definition 2.1 for zeros $\vec{\alpha^*}$ and $\vec{\alpha^0}$ the following inequalities are satisfied:

$$L_{TSP}(M_1) \leq L_{TSP}(M_2), \tag{9}$$

$$L_{TSP}(M_1 \cup H) \leq L_{TSP}(M_2 \setminus H). \tag{10}$$

Since, the graph is metric (splittable), according to Statement 2.1 the following conditions are valid:

$$L_{TSP}(M_1) \leq L_{TSP}(M_1 \cup H), \tag{11}$$

$$L_{TSP}(M_2 \setminus H) \leq L_{TSP}(M_2). \tag{12}$$

Hence, we immediately obtain:

$$\min\{L_{TSP}(M_1 \cup H), L_{TSP}(M_2 \setminus H)\} \leq \min\{L_{TSP}(M_1), L_{TSP}(M_2)\}. \tag{13}$$

Since, by the proposition, the value of expression

$$\min\{L_{TSP}(M_1), L_{TSP}(M_2)\}$$

is minimal for all bi-partitions of set V, the following equality is satisfied:

$$\begin{aligned} \min\{L_{TSP}(M_1 \cup H), L_{TSP}(M_2 \setminus H)\} \\ = \min\{L_{TSP}(M_1), L_{TSP}(M_2)\}. \end{aligned} \tag{14}$$

Therefore, for characteristic function $f(\vec{\alpha})$ there exists upper zero $\vec{\alpha}^*$ corresponding to an optimal solution of No-Depot Min-Max 2-TSP problem (4)-(6). Opposite vector $\vec{\alpha}^{**} = 1 - \vec{\alpha}^*$ is a lower one of the characteristic function; this vector corresponds to another optimal solution of the problem.

□

According to Theorem 2.1 each No-Depot Min-Max 2-TSP problem has a corresponding self-dual monotonic Boolean function. In the following sections we consider the reverse question:

What Boolean function has a corresponding No-Depot Min-Max 2-TSP, for which this function is characteristic?

Below we demonstrate that for every threshold self-dual monotonic function it is possible to find a No-Depot Min-Max 2-TSP, for which the considered function is characteristic. For the case discussed in the next sections the developed graph is splittable.

3. Threshold Characteristic Function

In this section we consider Boolean functions defined on set $\{-1, 1\}^n$. This definition of Boolean function is different from the standard one when the function is defined on $\{0, 1\}^n$. We use this format to emphasize specific properties

of the considered problem. Moreover, all the necessary properties of standard Boolean function remain valid for this class of functions. We identify every Boolean vector $\vec{\beta} = (\beta_1, \ldots, \beta_n)$ by vector $\| \beta_1, \ldots, \beta_n \|$ in linear space L_n.

A set of vectors on which $f(\vec{\beta}) = 0$ is called *a set of zeros* of function $f(\vec{\beta})$, and is denoted by $f^{-1}(0)$. *Set of ones* $f^{-1}(1)$ is defined similarly. Boolean function is called *threshold* [6], if there exists a set of real numbers $x_1, x_2, \ldots, x_n, c$, so that linear inequality

$$x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n \leq c$$

holds for Boolean vector $\vec{\beta} = (\beta_1, \ldots, \beta_n)$ *iff* $f(\vec{\beta}) = 0$. Thus, for threshold Boolean function $f(\vec{\beta})$ sets $f^{-1}(0)$ and $f^{-1}(1)$ are separated by hyperplane

$$x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n = c,$$

which is called *a threshold hyperplane* for threshold Boolean function $f(\vec{\beta})$. A hyperplane is called *a central hyperplane* if the coordinate origin belongs to this hyperplane.

The following statement makes a connection between self-dual threshold Boolean functions and central hyperplanes.

Statement 3.1. *Threshold Boolean function $f(\vec{\beta})$ is self-dual if and only if there exists a central threshold hyperplane for this function.*

Indeed, presence of a central threshold hyperplane implies self-duality of $f(\vec{\beta})$. Suppose now that threshold hyperplane $x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n = c$ of self-dual Boolean function $f(\vec{\beta})$ is not passing through the coordinate origin. Because of the self-duality, sets $f^{-1}(0)$ and $f^{-1}(1)$ are symmetric to each other. Consequently, hyperplane $x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n = -c$, which is symmetric to the original threshold hyperplane, is also a threshold hyperplane. Since the considered hyperplanes are threshold at the same time, the following conditions are valid:

$$\text{if } f(\vec{\beta}) = 1 \text{ then } x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n \geq c;$$

$$\text{if } f(\vec{\beta}) = 0 \text{ then } x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n \leq -c.$$

Consequently, central hyperplane

$$x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n = 0$$

is a threshold hyperplane for Boolean function $f(\vec{\beta})$. Therefore, we obtained a central threshold hyperplane that divides linear space L_n into two areas $A_0 = \{\vec{\beta} : x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n < 0\}$ and $A_1 = \{\vec{\beta} : x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n > 0\}$.

Hence, the next statement follows directly:

Statement 3.2. *$f^{-1}(0) \subset A_0$ and $f^{-1}(1) \subset A_1$.*

Indeed, because of self-duality, neither zero, nor one of Boolean function $f(\vec{\beta})$ belongs to the central threshold hyperplane. Therefore, all zeros of Boolean function $f(\vec{\beta})$ belong to area A_0, whereas all ones of the function belong to area A_1.

Definition 3.1. *Zero $(\beta_1, \ldots, \beta_{i-1}, -1, \beta_{i+1}, \ldots, \beta_n)$ of monotonic Boolean function $f(\vec{\beta})$ is the i^{th} frontier zero, if $(\beta_1, \ldots, \beta_{i-1}, +1, \beta_{i+1}, \ldots, \beta_n)$ is a one of function $f(\vec{\beta})$. One $(\beta_1, \ldots, \beta_{i-1}, +1, \beta_{i+1}, \ldots, \beta_n)$ of monotonic Boolean function $f(\vec{\beta})$ is the i^{th} frontier one, if $(\beta_1, \ldots, \beta_{i-1}, -1, \beta_{i+1}, \ldots, \beta_n)$ is a zero of function $f(\vec{\beta})$.*

The next statement follows immediately:

Statement 3.3. *For a monotonic self-dual Boolean function there exists at least one frontier zero, and at least one frontier one.*

Definition 3.2. *Function $f(\vec{\beta})$ is essentially independent of the i^{th} variable, if for any $\beta_1, \ldots, \beta_{i-1}, \beta_{i+1}, \ldots, \beta_n \in \{-1, +1\}$*

$$f(\beta_1, \ldots, \beta_{i-1}, -1, \beta_{i+1}, \ldots, \beta_n) = f(\beta_1, \ldots, \beta_{i-1}, +1, \beta_{i+1}, \ldots, \beta_n), \tag{15}$$

otherwise, function $f(\vec{\beta})$ is essentially dependent on the i^{th} variable,

Statement 3.4. *Monotonic Boolean function $f(\vec{\beta})$ is essentially dependent on the i^{th} variable if and only if this function has the i^{th} frontier zero and the i^{th} frontier one.*

This statement immediately follows from Definitions 3.1 and 3.2.

Theorem 3.1. *For any threshold self-dual Boolean function $f(\vec{\beta})$, essentially dependent on all its variables and defined by threshold hyperplane $x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n = c$ any nontrivial vector $\| a_1, \ldots, a_n \|$, $a_1, \ldots, a_n \geq 0$ belongs to area $A_1 = \{\beta : x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n > 0\}$, whereas the opposite vector $\| -a_1, \ldots, -a_n \|$ belongs to area $A_0 = \{\beta : x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n < 0\}$.*

Proof. At first, we show the validity of the theorem for vector $\| 1, 0, \ldots, 0 \|$. Denote $R = \{\beta : x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n = 0\}$. According to Statement 3.1, R is a central threshold hyperplane for Boolean function $f(\vec{\beta})$.

Suppose that vector $\| 1, 0, \ldots, 0 \|$ does not belong to A_1. Therefore,

$$\| 1, 0, \ldots, 0 \| \in A_0 \cup R.$$

If $\| 1,0,\ldots,0 \| \in R$ then $x_1 = 0$, and hyperplane $x_2\beta_2 + \ldots + x_n\beta_n = 0$ is a threshold hyperplane of function $f(\vec{\beta})$. Hence, function $f(\vec{\beta})$ does not depend on variable β_1, that contradicts the condition of the essential dependency. Therefore

$$\| 1,0,\ldots,0 \| \in A_0.$$

Since function $f(\vec{\beta})$ is essentially dependent on all its variables, due to Statement 3.4, this function has the 1-st frontier zero $\| -1,\beta_2,\ldots,\beta_n \| \in A_0$.

Since A_0 is a conical set, and $\| 1,0,\ldots,0 \|$ and $\| -1,\beta_2,\ldots,\beta_n \|$ belong to that set,

$$(\| 1,0,\ldots,0 \| + 2 \| -1,\beta_2,\ldots,\beta_n \|) \in A_0.$$

By Definition 3.1,

$$\| 1,\beta_2,\ldots,\beta_n \| = \| 1,0,\ldots,0 \| + 2 \| -1,\beta_2,\ldots,\beta_n \|$$

is a frontier one, and, consequently, belongs to set A_1. Therefore,

$$A_1 \cup A_0 \neq \emptyset.$$

Thus, we obtain a contradiction of what we have assumed. Therefore,

$$\| 1,0,\ldots,0 \| \in A_1.$$

Similarly, the statement can be proved for any other unit vector

$$\| 0,\ldots,0,1,0,\ldots,0 \| .$$

Since A_1 is a conical set, any nontrivial linear conical combination (combination with non-negative coefficients) of its elements belongs to this set. Therefore, for any nontrivial combination of nonnegative numbers $a_1,\ldots,a_n$,

$$\| -a_1,\ldots,-a_n \| \in A_0$$

and

$$\| a_1,\ldots,a_n \| \in A_1.$$

□

The following statement is an immediate inference of the theorem:

Statement 3.5. *If threshold self-dual Boolean function $f(\vec{\beta})$ is essentially dependent on all its variables, and $x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n = c$ is the threshold hyperplane of the function $f(\vec{\beta})$, then $x_i > 0$ for any $i = 1,\ldots,n$.*

Indeed, according to Theorem 3.1 , the vectors of form $\| 0,\ldots,0,1,0,\ldots,0 \|$ satisfy inequality $x_1\beta_1 + x_2\beta_2 + \ldots + x_n\beta_n \geq 0$, that proves the statement.

4. Constant Graphs

Definition 4.1. *Symmetric graph $G(V, E)$ is a constant graph if there exists a set of numbers $B = \{b_i\}_{i=1}^{i=n}$, so that for every pair of vertexes i and j of graph $G(V, E)$, the length of a connecting edge is $d(i, j) = b_i + b_j$. Set $B = \{b_i\}_{i=1}^{i=n}$ is a defining set for constant graph $G(V, E)$.*

The TSP considered on a constant graph belongs to the class of Constant Discrete Programming Problems, which are studied, for example, in [3].

Statement 4.1. *For constant symmetric graph $G(V, E)$ the length of a cycle does not depend on the order of the visits to the vertexes of the cycle.*

Indeed, the length of cycle $c = (j_1 \to j_2 \to \ldots \to j_k \to j_1)$ is determined by

$$L(c) = \sum_{i=1}^{k-1} d(j_i, j_{i+1}) + d(j_k, j_1) \tag{16}$$

$$= \sum_{i=1}^{k-1} (b_{j_i} + b_{j_{i+1}}) + bj_k + bj_1 = 2\sum_{i=1}^{k-1} b_{j_i} = 2\sum_{j_i \in c} b_{j_i}. \tag{17}$$

Moreover, it can be proved that if for a complete graph the length of a cycle does not depend on the order of the visits to the vertexes of the cycle, the graph is constant [3]. As a direct consequence of Statement 4.1, TSP is a trivial problem for a constant graph. Indeed, every tour in that graph has the same length.

Statement 4.2. *For constant graph $G(V, E)$ with non-negative elements of defining set $B = \{b_i\}_{i=1}^{i=n}$, the triangle inequality holds for all triplets of vertexes of that graph.*

Indeed,

$$d(i, j) + d(j, k) = (b_i + b_j) + (b_j + b_k)$$
$$= (b_i + b_k) + 2b_j = d(i, k) + 2b_j \geq d(i, k)$$

5. Interpretation of Threshold Self-Dual Monotonic Boolean Functions

Consider symmetric constant graph $G(V, E)$ with nonnegative elements in its defining set $B = \{b_i\}_{i=1}^{i=n}$. As it has been shown in the previous section, the triangle inequality holds for any triplet of the graph vertexes, and the length of a cycle does not depend on the order of the visits to the vertexes.

Assuming that graph $G(V, E)$ is splittable, consider No-Depot Min-Max 2-TSP assigned to this graph. The characteristic function of constant graph $G(V, E)$ is defined by the following rule:

$$f(\vec{\beta}) = \begin{cases} 1 & \text{if } 2\sum_{i\in M_1} b_i \geq 2\sum_{i\in M_2} b_i \\ 0 & \text{otherwise} \end{cases}, \tag{18}$$

where

$$\beta_i = \begin{cases} 0 & \text{if } i \in M_1 \\ 1 & \text{if } i \in M_2 \end{cases}, \tag{19}$$

$$M_1 \cup M_2 = V,\ M_1 \cap M_2 = \emptyset.$$

Using the vector notations introduced above and an assumption that graph $G(V, E)$ is splittable, the definition of function $f(\vec{\beta})$ can be rewritten in the following way:

$$f(\vec{\beta}) = \begin{cases} 1 & \text{if } \vec{\beta}^T\vec{x} > 0 \\ 0 & \text{if } \vec{\beta}^T\vec{x} < 0 \end{cases}, \tag{20}$$

where

$$\vec{\beta} = \|\beta_1, \ldots, \beta_n\|, \vec{x} = \|x_1, \ldots, x_n\|.$$

Since characteristic function $f(\vec{\beta})$ is monotonic, the determination of all its lower ones is sufficient to completely describe the function. Let F_1 denote the set of lower ones, and F_0 denote the set of upper zeros of function $f(\vec{\beta})$. Due to the self-duality of function $f(\vec{\beta})$ the following statement is valid for splittable graph $G(V, E)$:

Statement 5.1. *$\vec{\beta} \in F_1 \Leftrightarrow -\vec{\beta} \in F_0$, that is, the vector opposite to a lower one is an upper zero, and vice versa.*

Definition 5.1. *A fundamental matrix of ones of monotonic Boolean function $f(\vec{\beta})$ is matrix D_1, constructed by the following rules: the first n rows form an identity matrix $n \times n$; the next rows are formed by all the vectors from F_1 arranged in binary increasing order. A fundamental matrix of zeros of monotonic Boolean function $f(\vec{\beta})$ is matrix D_0, constructed by the following rules: the first n rows form a negative identity matrix $n \times n$; the next rows are formed by all the vectors from F_0 arranged in binary decreasing order.*

Due to self-duality of function $f(\vec{\beta})$, $D_1 = -D_0$.

Statement 5.2. *Any one $\vec{\beta}_1$ of monotonic Boolean function $f(\vec{\beta})$ can be represented as $\vec{\beta}_1 = \|b_1, \ldots, b_m\| \times D_1$; any zero $\vec{\beta}_0$ of monotonic Boolean function $f(\vec{\beta})$ can be represented as $\vec{\beta}_0 = \|b_1, \ldots, b_m\| \times D_0$, where $b_i \in \{0, 2\}$, $i = 1, \ldots, n$, and $b_i \in \{0, 1\}$, $i = n+1, \ldots, m$, and m is the number of rows in fundamental matrix D_1 (D_0).*

Indeed, one $\vec{\beta}_1$ of function $f(\vec{\beta})$ is generated by some lower one $\vec{\beta}_1^*$. Suppose that $\vec{\beta}_1^*$ is represented by the k^{th} row of fundamental matrix D_1. Then

$$\vec{\beta}_1 = \|b_1, \ldots, b_n, 0, \ldots, 0, b_k = 1, 0, \ldots, 0\| \times D_1, \tag{21}$$

where

$$b_i = \begin{cases} 2 & \text{if } \vec{\beta}_1 \text{ differs from } \vec{\beta}_1^* \text{ in the } i^{\text{th}} \text{ component} \\ 0 & \text{if } \vec{\beta}_1 \text{ conicides with } \vec{\beta}_1^* \text{ in the } i^{\text{th}} \text{ component} \end{cases} \tag{22}$$

Similarly, it can be shown for zero $\vec{\beta}_0$ of function $f(\vec{\beta})$.

For the rest of the section, notation $D \times \vec{x} > 0$ ($D \times \vec{x} < 0$) means that resultant vector $\vec{v} = D \times \vec{x}$ has only positive (negative) components.

Theorem 5.1. *Self-dual monotonic Boolean function $f(\vec{\beta})$ essentially dependent on all its variables is an threshold if and only if there exists a vector $\vec{x}$, for which $D_1 \times \vec{x} > 0$, where D_1 is a fundamental matrix of ones of function $f(\vec{\beta})$.*

Proof. **Existence:** Due to Statement 3.1 self-dual threshold function $f(\vec{\beta})$ has central threshold hyperplane $(\vec{\beta}, \vec{x}) = 0$, and for any one $\vec{\beta}_1$ of function $f(\vec{\beta})$

$$(\vec{\beta}_1, \vec{x}) > 0.$$

According to Statement 3.5, all components of vector $\vec{x}$ are positive. Since the rows of matrix D_1 are either rows of the unit matrix, or lower ones of function $f(\vec{\beta})$, the following inequality is satisfied:

$$D_1 \times \vec{x} > 0.$$

Self-Duality: Suppose that there exists vector $\vec{x}$, so that $D_1 \times \vec{x} > 0$. Consider any one $\vec{\beta}_1$ of function $f(\vec{\beta})$. Due to Statement 5.2, $\vec{\beta}_1 = \|b_1, \ldots, b_m\| \times D_1$, where $b_i \in \{0, 2\}$, $i = 1, \ldots, n$, and $b_i \in \{0, 1\}$, $i = n+1, \ldots, m$. Denote $\vec{v} = D_1 \times \vec{x}$. Since $v_i > 0$ and $b_i \geq 0$ for any $i = 1, \ldots, m$, and there exist i^*, for which $b_{i^*} > 0$, then

$$(\vec{\beta}_1, \vec{x}) = (\vec{b} \times D_1, \vec{x}) = (\vec{b}, (D_1 \times \vec{x})^T) \geq$$

$$\geq b_{i^*} \times \min_i v_i > 0.$$

Hence, for one $\vec{\beta}_1$ of function $f(\vec{\beta})$

$$(\vec{\beta}_1, \vec{x}) > 0.$$

Since function $f(\vec{\beta})$ is self-dual, for any zero $\vec{\beta}_0$ of function $f(\vec{\beta})$

$$(\vec{\beta}_0, \vec{x}) < 0.$$

Therefore, function $f(\vec{\beta})$ is a threshold one with hyperplane $(\vec{\beta}, \vec{x}) = 0$. □

Theorem 5.2. *For any threshold self-dual monotonic Boolean function $f(\vec{\beta})$ essentially dependent on all its variables there exists a graph, for which function $f(\vec{\beta})$ is characteristic.*

Proof. Consider threshold self-dual monotonic Boolean function $f(\vec{\beta})$, which is essentially dependent on all its variables. Let D_1 be a fundamental matrix of ones of function $f(\vec{\beta})$. Due to Theorem 5.1, there exists vector $\vec{x}$ for which $D_1 \times \vec{x} > 0$. Since the first n rows of matrix D_1 form an identity matrix, the first n inequalities in $D_1 \times \vec{x} > 0$ set $\vec{x} > 0$. The rest of the rows of matrix D_1, by definition, correspond to lower ones of function $f(\vec{\beta})$. Define constant graph $G(V, E)$ by defining set $B = \{b_i = x_i, i = 1, \ldots, n\}$. The length of edge (i, j) is $d(i, j) = b_i + b_j = x_i + x_j$. Consider No-Depot Min-Max 2-TSP on this graph. Let $\widetilde{f(\vec{\beta})}$ be a characteristic function of the graph. Due to Statements 2.1 and 2.2 function $\widetilde{f(\vec{\beta})}$ is monotonic and self-dual. Therefore, the following condition satisfies each one $\vec{\beta_1}$ (and each zero $\vec{\beta_0} = -\vec{\beta_1}$) of function $f(\vec{\beta})$:

$$(\vec{\beta_1}, \vec{x}) > 0.$$

Since $D_1 \times \vec{x} > 0$ and all the lower ones of function $f(\vec{\beta})$ are among the rows of matrix D_1, every lower one of Boolean function $f(\vec{\beta})$ is a one of function $\widetilde{f(\vec{\beta})}$. Because of monotonicity and self-duality of functions $f(\vec{\beta})$ and $\widetilde{f(\vec{\beta})}$, and the fact, that any monotonic Boolean function is completely defined by the set of lower ones, the equation

$$f(\vec{\beta}) = \widetilde{f(\vec{\beta})}$$

is valid for any binary $\vec{\beta}$. □

Note, that in the proof, constructed graph $G(V, E)$ is splittable.

Consider the case when Boolean threshold function $f(\vec{\beta})$ is independent of some of its variables. The following statement is valid for this function:

Statement 5.3. *If threshold function $f(\vec{\beta})$ is independent of the i^{th} variable, and $(\vec{\beta}, \vec{x}) = c$ is a threshold hyperplane for this function, then $(\vec{\beta}, \vec{y}) = c$, where*

$$y_j = \begin{cases} 0 & j = i \\ x_j & j \neq i \end{cases}$$

is also a threshold hyperplane of function $f(\vec{\beta})$

Indeed, consider a one $\vec{\beta_1}$ of function $f(\vec{\beta})$. It satisfies $(\vec{\beta_1}, \vec{x}) \geq c$. Since function $f(\vec{\beta})$ is independent of the i^{th} variable, there exists a one $\vec{\beta_1}'$ of the

function that differs from $\vec{\beta}$ by only the i^{th} variable. Therefore, $(\vec{\beta_1}', \vec{x}) \geq c$. Adding inequalities $(\vec{\beta_1}, \vec{x}) \geq c$ and $(\vec{\beta_1}', \vec{x}) \geq c$, we get that $\vec{\beta_1}$ satisfies $(\vec{\beta_1}, \vec{y}) \geq c$. Similarly, any zero $\vec{\beta_0}$ of function $f(\vec{\beta})$ satisfies $(\vec{\beta_0}, \vec{y}) \leq c$. Therefore, $(\vec{\beta}, \vec{y}) = c$ is a threshold hyperplane of function $f(\vec{\beta})$. The reverse statement appears straightaway:

Statement 5.4. *If threshold Boolean function $f(\vec{\beta})$ has threshold hyperplane $(\vec{\beta}, \vec{x}) = c$, and $x_i = 0$, then function $f(\vec{\beta})$ is independent of the i^{th} variable.*

Definition 5.2. *Central threshold hyperplane $(\vec{\beta}, \vec{x}) = 0$ of threshold Boolean function $f(\vec{\beta})$ is a reduced hyperplane, if for every variable, of which function $f(\vec{\beta})$ is independent, the corresponding component of vector $\vec{x}$ is zero.*

A central threshold hyperplane of a Boolean function essentially dependent on all its variables is a simple case of a reduced hyperplane. Due to Statements 3.1 and 5.3 and the definition of central threshold hyperplane, the following statement is valid:

Statement 5.5. *For any threshold self-dual Boolean function there exists a reduced central threshold hyperplane.*

We are ready to prove the main result of the chapter.

Theorem 5.3. *For any threshold self-dual monotonic Boolean function $f(\vec{\beta})$ there exists a graph, for which function $f(\vec{\beta})$ is characteristic.*

Proof. The case with function $f(\vec{\beta})$ essentially dependent on all its variables has been considered in Theorem 5.2. Now we consider the case when function $f(\vec{\beta})$ is independent of some of its variables. Let $K_{indep} = \{1, \ldots, k\}$ be a set of indexes of variables, on which the function $f(\vec{\beta})$ does not depend, and $K_{dep} = \{k+1, \ldots, n\}$ be a set of indexes of the variables, on which function $f(\vec{\beta})$ essentially depends. According to Statement 5.5, there exists reduced central threshold hyperplane $(\vec{\beta}, \vec{y}) = 0$ for function $f(\vec{\beta})$. Define constant graph $G(V, E)$ by defining set $B = \{b_i = y_i \; i = 1, \ldots, n\}$. The length of edge (i, j) is

$$d(i, j) = b_i + b_j = y_i + y_j.$$

For any variable, on which function $f(\vec{\beta})$ does not depend, the corresponding component of $\vec{y}$ is zero. In compliance with the proof of Theorem 5.2, Boolean function $f(\vec{\beta})$ is a characteristic function for No-Depot Min-Max 2-TSP considered on graph $G(V, E)$. □

The same as in the proof of Theorem 5.2, the obtained graph is splittable.

6. Conclusion

The Multiple Traveling Salesmen Problem has many variations. In this work we studied a No-Depot Min-Max formulation of 2-Traveling Salesmen Problem. We introduced a characteristic function for this class of problems. This Boolean function is monotonic and self-dual for complete graphs with metric distances. For an arbitrary monotonic threshold self-dual Boolean function we have proven existence of a No-Depot Min-Max 2-Traveling Salesmen Problem, for which this function is characteristic.

References

[1] M. Bellmore and S. Hong. Transformation of multisalesmen problem to the standard traveling salesman problem. *J. Assoc. Comput. Machinery*, 21:500–504, 1974.

[2] P. M. Franca, M. Gendreau, G. Laporte, and F. M. Muller. The m-traveling salesman problem with minmax objective. *Transportation Science*, 29:267–275, 1995.

[3] E. Ya. Gabovich and I. I. Melamed. On constant discrete programming problems. *Discrete Applied Mathematics*, 2:193–205, 1980.

[4] E. Giust. *Optimisation de Tourness de Vehicules. Application a la Distribution de Gaz.* M.Sc. Dissertation. Facultes Universitaires Notre-Dame de la Paix, Namur, Belgium, 1992.

[5] M. R. Rao. A note on the multiple traveling salesmen problem. *Operations Research*, 28:628–632, 1980.

[6] Yu. A. Zuev. Threshold functions and threshold interpretations of Boolean functions. *Matematicheskie Voprosy Kibernetiki*, 5:5–61, 1994. In Russian.